Meta Heuristics based Machine Learning and Neural Mass Modelling Allied to Brain Machine Interface

A Dissertation Presented

by

Elham Zareian

in

College of Science

School of Engineering

University of Lincoln

A thesis submitted in partial fulfilment for the degree of

Doctor of Philosophy

April 2021

To my wonderful parents, for their unconditional love and support.

Declaration

I hereby declare that except where specific reference is made to the work of others, the contents of this dissertation are original and have not been submitted in whole or in part for consideration for any other degree or qualification in this, or any other university. This dissertation is my own work and contains nothing which is the outcome of work done in collaboration with others, except as specified in the text and Acknowledgements. This dissertation contains fewer than 65000 words including appendices, bibliography, footnotes, tables and equations and has fewer than 50 figures.

A Dissertation Presented

by

Elham Zareian

April 2021

Abstract

New understanding of the brain function and increasing availability of low-cost-non-invasive electroencephalograms (EEGs) recording devices have made brain-computer-interface (BCI) as an alternative option to augmentation of human capabilities by providing a new non-muscular channel for sending commands, which could be used to activate electronic or mechanical devices based on modulation of thoughts. In this project, our emphasis will be on how to develop such a BCI using fuzzy rule-based systems (FRBSs), metaheuristics and Neural Mass Models (NMMs). In particular, we treat the BCI system as an integrated problem consisting of mathematical modelling, machine learning and classification. Four main steps are involved in designing a BCI system: 1) data acquisition, 2) feature extraction, 3) classification and 4) transferring the classification outcome into control commands for extended peripheral capability. Our focus has been placed on the first three steps.

This research project aims to investigate and develop a novel BCI framework encompassing classification based on machine learning, optimisation and neural mass modelling. The primary aim in this project is to bridge the gap of these three different areas in a bid to design a more reliable and accurate communication path between the brain and external world.

To achieve this goal, the following objectives have been investigated: 1) Steady-State Visual Evoked Potential (SSVEP) EEG data are collected from human subjects and pre-processed; 2) Feature extraction procedure is implemented to detect and quantify the characteristics of brain

activities which indicates the intention of the subject.; 3) a classification mechanism called an Immune Inspired Multi-Objective Fuzzy Modelling Classification algorithm (IMOFM-C), is adapted as a binary classification approach for classifying binary EEG data. Then, the DDAG-Distance aggregation approach is proposed to aggregate the outcomes of IMOFM-C based binary classifiers for multi-class classification; 4) building on IMOFM-C, a preference-based ensemble classification framework known as IMOFM-CP is proposed to enhance the convergence performance and diversity of each individual component classifier, leading to an improved overall classification accuracy of multi-class EEG data; and 5) finally a robust parameterising approach which combines a single-objective GA and a clustering algorithm with a set of newly devised objective and penalty functions is proposed to obtain robust sets of synaptic connectivity parameters of a thalamic neural mass model (NMM). The parametrisation approach aims to cope with nonlinearity nature normally involved in describing multifarious features of brain signals.

Acknowledgements

I would like to thank the following people for their assistance in the completion of this project:

First of and foremost I am extremely grateful to my supervisors, Dr Jun Chen, Professor Timothy Gordon and Dr. Basabdatta Sen Bhattacharya, for their invaluable advice, continuous support and patience during my PhD study. Their Immense knowledge and encouragement helped me in all the time of research and writing of this thesis.

I am also thankful to Dr. Louise O'Hare, for her treasured help with data collection at school of Psychology and her excellent advice that greatly improved my data collection experiments.

A huge thanks goes to my dear friends, Dr Michal Weiszer, and Mosab Bazargani for their insightful suggestions and their great support throughout my PhD study.

I would like to express the deepest gratitude to my beloved parents, for all their encouragement, care, support and the scarifies that they have made for me through my entire life. I would also like to thank my lovely brother, for his priceless and kind support. A special thanks goes to my uncle and auntie, who supported me to come to this country and pursue my study.

Lastly, I would like to thank school of Engineering at University of Lincoln, School of Engineering and Materials Science at Queen Mary University of London, and Anglian Water Company, for their support both financially and academically.

Related Publications

During the course of PhD. study, a number of papers represented my ongoing research are produced. The following is the list of them.

Zareian, E., Kathpalia, V., Chen, J., and Smith, T., (2014): A cost effective approach for the practical realisation of a demonstration platform for brain machine interface. Second International Conference on Advances in Computing, Communication and Information Technology-CCIT 2014, November 2014, Birmingham, United Kingdom.

Zareian, E., Chen, J., and Bhattacharya, B. Sen., (2016): A Robust Evolutionary Optimisation Approach for Parameterising a Neural Mass Model. 25th International Conference on Artificial Neural Networks- ICANN 2016, September 2016, Barcelona, Spain.

Chen, J., Weiszer, M., Zareian, E., Mahfouf, M., Obajemu, O., (2017): Multi-objective fuzzy rule-based prediction and uncertainty quantification of aircraft taxi time. 2017 IEEE International Conference on Intelligent Transportation Systems (ITSC), 16-19 Oct. 2017, Yokohama, Japan.

Zareian, E., Chen, J., O'Hare, L., Bhattacharya, B. Sen., and Gordon, T., (2018): Interpretable Fuzzy Rule-based Systems for Classification of Multi-Class EEG Data. 2018 IEEE International Conference on Systems, Man, and Cybernetics (SMC), 7 - 10 Oct. 2018, Miyazaki, Japan.

Contents

List of Figures

List of Tables

Abbreviations

AAO	**A**ll-**A**t-**O**nce
AD	**A**lzheimer **D**isease
ANFIS	**A**daptive **N**euro **F**uzzy **I**nference **S**ystem
ANN	**A**rtificial **N**eural **N**etworks
AR	**A**utoregressive **M**odelling
BCI	**B**rain **C**omputer **I**nterface
BEP	**B**ack-**E**rror **P**ropagation
CA	**C**ornu **A**mmonis
CKF	**C**ubature **K**alman **F**ilter
CNS	**C**entral **N**ervous **S**ystem
CWT	**C**ontinuous **W**avelet **T**ransform
DDAG	**D**ecision **D**irected **A**cyclic **G**raph
DDAG-Dist	**D**ecision **D**irected **A**cyclic **G**raph **D**istance
DFT	**D**iscrete **F**ourier **T**ransform
DIVACE	**Div**erse and **Ac**curate **E**nsemble learning algorithm
DWT	**D**iscrete **W**avelet **T**ransform
EA	**E**volutionary **A**lgorithm
EEG	**E**lectroencephalogram
ECG	**E**lectrocardiographic
eIN	**E**xcitatory **In**terneurons
EKF	**E**xtended **K**alman **F**ilter
EMG	**E**lectromyographic

EOG	**E**lectrooculographic
EPSP	**E**xcitatory **P**ost-**S**ynaptic **P**otential
ERP	**E**vent **R**elated **P**otentials
ERD	**E**vent-**R**elated-**D**esynchronisation
ERS	**E**vent-**R**elated-**S**ynchronisation
FFT	**F**ast **F**ourier **T**ransform
FIR	**F**inite **I**mpulse **R**esponse
FIS	**F**uzzy **I**nference **S**ystem
fIN	**F**ast **i**nhibitory **In**terneurons
FN	**F**alse **N**egative
FP	**F**alse **P**ositive
FRBS	**F**uzzy **R**ule-**B**ased **S**ystem
GA	**G**enetic **A**lgorithm
GABA	**G**amma-**A**mino**b**utyric-**A**cid
GD	**G**enerational **D**istance
GP	**G**enetic **P**rogramming
HV	**H**yper**v**olume
IFR	**I**nfinite **I**mpulse **R**esponse
IGA	**I**diopathic **G**eneralised **E**pilepsies
IGD	**I**nverted **G**enerational **D**istance
IMOFM-C	**I**mmune inspired **M**ulti-**O**bjective **F**uzzy **M**odelling **C**lassification
IMOFM-CP	**I**mmune inspired **M**ulti-**O**bjective **F**uzzy **M**odelling **C**lassification Preference- based
IN	**I**nterneurons
IPSP	**I**nhibitory **P**ost-**S**ynaptic **P**otential

KF	**K**alman **F**ilter
LDA	**L**inear **D**iscriminant **A**nalysis
LGN	**L**ateral **G**eniculate **N**ucleus
MLP	**M**ulti-**L**ayer **P**erceptron
MOEA	**M**ulti-**O**bjective **E**volutionary **A**lgorithm
MOP	**M**ulti-objective **O**ptimisation **P**roblem
NCL	**N**egative **C**orrelation **L**earning
NMM	**N**eural **M**ass **M**odel
OAA	**O**ne-**A**gainst-**A**ll
OAO	**O**ne-**A**gainst-**O**ne
PAIA	A **P**opulation **A**daptive based **I**mmune **A**lgorithm
PFC	**P**airwise **F**ailure **C**rediting
PNS	**P**eripheral **N**ervous **S**ystem
PSD	**P**ower **S**pectrum **D**ensities
PSP	**P**ost-**S**ynaptic **P**otential
PY	**Py**ramidal cells
SE	**S**tatus **E**pilepticus
sIN	**S**low **i**nhibitory **In**terneurons
SOP	**S**ingle-objective **O**ptimisation **P**roblems
SSVEP	**S**teady-**S**tate **V**isual **E**voked **P**otential
STFT	**S**hort-**T**ime **F**ourier **T**ransform
SVM	**S**upport **V**ector **M**achine
SW	**S**pike and **W**ave
TCR	**T**halamic **C**ortica**l** **R**elay
TCT	**T**halamo-**C**ortico-Thalamic
TN	**T**rue **N**egative

TP	**T**rue **P**ositive
TRN	**T**halamic **R**eticular **N**ucleus
UKF	**U**nscented **K**alman **F**ilter
WT	**W**avelet **T**ransform

Chapter 1

Introduction

1.1 General Background

New understanding of the brain function and increasing availability of low-cost-non-invasive electroencephalograms (EEGs) recording devices have made brain-computer-interface (BCI) as an attractive option in order to augment human capabilities by providing a new non-muscular channel for sending commands which could be used to activate electronic or mechanical devices based on modulation of thoughts. In this project, the emphasis will be on how to develop such a BCI using fuzzy rule-based systems (FRBSs), metaheuristics and Neural Mass Models (NMMs). In particular, we will treat the BCI system as an integrated problem consisting of mathematical modelling, machine learning and classification. Four main steps are involved in designing a BCI system: 1) data acquisition, 2) feature extraction, 3) classification and 4) transferring the classification outcome into control commands for extended peripheral capability. Our focus has been placed on the first three steps. Although there exist a wide range of approaches addressing BCI, a systematic way of designing an ensemble classifier using preference-based multi-objective metaheuristics and FRBSs has not been found in the previous studies. Furthermore, the idea of using a set of biologically plausible parameters offered by NMMs for classification purpose has not been attempted so far. In light of the above, in addition to specifically designed data acquisition and feature extraction steps, this project has also been

centred around three main topics, viz. FRBS-based classifier for multiclass classification, multi-objective ensemble classification and parameterising NMMs.

1.2 Project Description

This research project aims to investigate and develop a novel BCI framework encompassing classification based on machine learning, optimisation and neural mass modelling. The primary aim in this project is to bridge the gap of these three different areas in a bid to design a more reliable and accurate communication path between the brain and external world.

Traditional ways of studying BCI often include acquiring appropriate brain signals and establishing classification models based on the features extracted from the acquired data (Mason and Birch 2003). Although the acquired signal types and features for classification are varied, they represent the state-of-the-art in BCI towards interfacing human intentions with the outside world. Different BCI applications often require different brain signals which are categorised in five frequency bands, namely a) Delta (<4 HZ), b) Theta (4-7 HZ), c) Alpha (8-15 HZ), d) Beta (16-31 HZ), and e) Gamma (32+ HZ) (Buzsáki 2006). Appropriate feature extraction methods (Mallat 1989, Al-Fahoum and Al-Fraihat 2014) have to be chosen in an ad-hoc manner in order to achieve satisfactory classification accuracy.

The ad-hoc way of choosing which features should be included for classification put a significant constraint on studying BCI as these features normally do not reflect the physiological mechanisms underlying brain activities. Therefore, these biologically implausible features are data dependent, i.e. prone to be sensitive to subjects and varying environmental conditions, inevitably leading to the requirement of a fairly large number of participating subjects and stringent experimental conditions. NMMs may help to overcome this problem by modelling the areas of the interest of the human brains. Furthermore, NMMs offer a set of biological

plausible connectivity parameters. Those biologically plausible parameters *de facto* offer an effective way of simulating various brain signals across different subjects under varying testing environments (e.g. with different noise and interruptions).

The significance of this modelling capability lies in: 1) speeding up the research progress by saving the time spent on recruiting and training the right subject, as well as on repeating the recording process until sufficient and satisfactory data are acquired; 2) providing more data under various experimental conditions which are not attainable in real experiments due to time, personal and environmental constraints.

There are two important facts that need to be taken into account while using neural mass models: 1) Choosing the right model for a particular application; there is an extensive literature review on using neural mass models; although mostly in medical applications (such as diagnosing epilepsy and seizure (Lopes da Silva et al. 1974), they provide direct references for simulating brain signals for non-medical applications (Petersen et al. 2014) .2) Parameterising neural mass models; depending on the particular application domain, part or all of the model's parameters need to be parameterised so that the output of the model reflects the underlying mechanisms of brain activities; the parametrised model should present certain robustness to the dynamic and changing environment both external and internal to the model (Paenke and Branke 2006, Salomon et al. 2014, Zareian, Chen, and Bhattacharya 2016)

Obtaining sets of optimal and robust connectivity parameters underpinning the model with its output reflecting experimental EEG data is prerequisite for a successful BCI application. The recorded EEG data in this case is used as a reference so that the model can capture the key features (e.g. the spikes, local minima between spikes, inflection points, and the number/ordering/positions of them that take place in a cycle (Nevado-Holgado et al. 2012)) of the measured signals, producing an output containing the same features. A data fitting procedure involves

the four main steps which are listed as follow: 1) signal pre-processing 2) segmentation 3) feature extraction, and finally 4) obtaining the optimal parameter sets for the neural mass model. As the connectivity parameters are governed by mathematical modelling of brain activities, they are not purely data-driven parameters. Therefore, obtaining the optimal parameters through the data fitting procedure offers a potentially alternative route of using the connectivity parameters as the direct features for classification purpose.

In designing a BCI system, the first three aforementioned data fitting steps are also the main steps at the classification stage. Therefore, the same steps are also adopted in the development of FRBS-based multiclass and ensemble classifiers. FRBS is an approach that combines both human knowledge and machine learning to account for complex systems' knowledge acquisition. To further improve the transparency and generalisation capability of FRBSs, FRBSs have been hybridised with different learning algorithms including neural networks (Jang 1993) (J. Chen and Mahfouf 2012) and multi-objective metaheuristics. A particular line of research focused on hybridising multi-objective metaheuristics with FRBSs aims to simultaneously improve models' transparency and generalisation capability (J. Chen and Mahfouf 2012). Multi-objective metaheuristics are very suitable for that task as it derives a set of Pareto solution by simultaneously evolving the structure and parameters of the FRBS. In this project, a particular multi-objective FRBS modelling approach (J. Chen and Mahfouf 2012) has been chosen for its good performance in generalisation and ability to express the knowledge of the elicited model. The approach was originally designed for regression problems but have been adapted for binary classification in this project.

BCI systems normally involve more than two control commands that are associated with corresponding brain activities, leading to a multi-class classification problem. Instead of having a single classifier that can relate respective brain activities to different control commands, it has

been shown that by decomposing a multi-class problem into a set of binary classification problems, the overall classification accuracy can be improved if appropriate aggregation methods are used at the end (Eichelberger and Sheng 2013). In light of this, we decomposed the multi-class classification problem into six binary classification problems corresponding to EEG signals subject to different stimulus at different frequencies and adopted an arrogation method to give the overall classification result.

As the multi-objective FRBS-based classification approach leads to a set of FRBS-based classifiers with varying structures and parameters, we further exploit that property to develop the adapted FRBS-based approach into an ensemble classification framework. It has been shown that as long as individual component classifiers are sufficiently different from each other and accurate enough, using an appropriate committee voting strategy will lead to an classification result that is no worse (often better) than the performance of each component classifier (Xin Yao and Yong Liu 1998). Therefore, the concept of preference-based search has been further incorporated into the learning process so that the search is focused only on the region that can produce a set of accurate enough FRBSs, resulting in a better convergence at the end of the search process (Yi et al. 2018). By taking into account diversity as one of the objectives, the proposed ensemble classification framework can simultaneously improve the diversity and classification accuracy of each individual component classifier, leading to a better overall classification accuracy.

In light of the above, the overarching aims of this project are:

1) Design experiments for extracting Steady-State Visual Evoked Potential (SSVEP) signals, which were in line with the requirement of Neural Mass Modelling.

2) Develop a feature extraction procedure to detect and quantify the characteristics of brain activities which indicates the intention of the subject.

3) Develop an ensemble classification mechanism based on multi-objective fuzzy rule-based systems and preference-based search algorithms in order to classify the feature vectors in the right classes with the aim to increase the classification accuracy rate.

4) Develop a robust NMMs parameterisation approach based on metaheuristics and clustering algorithms and a data fitting procedure using the extracted features in 2).

5) Investigate the difference in the identified connectivity parameters subject to different stimulus.

1.3 Original Contributions

The original contributions of this thesis are of the followings:

- A classification mechanism called an Immune Inspired Multi-Objective Fuzzy Modelling Classification algorithm (IMOFM-C), which was originally designed for regression problems, is adapted as binary classification approach for classifying multi-class Electroencephalogram (EEG) data.

- An aggregation method called Decision Directed Acyclic Graph Distance (DDAG-Distance) is proposed to aggregate the outcomes of IMOFM-C binary classifiers.

- Building on IMOFM-C, a preference-based ensemble classification framework known as IMOFM-CP is proposed to enhance the convergence performance and diversity of each individual component classifier, leading to an improved overall classification accuracy of multi-class EEG data.

- The proposed ensemble classification mechanism has been rigorously tested with benchmark problems and has been applied to the classification of Steady-State Visual

Evoked Potential (SSVEP) EEG data of 4 human subjects, showing improved classification accuracy against that of individual component classifier.

- A robust parameterising approach which combines a single-objective GA and a clustering algorithm with a set of newly devised objective and penalty functions is proposed to obtain robust sets of synaptic connectivity parameters of a thalamic neural mass model (NMM). The parametrisation approach aims to cope with nonlinearity nature normally involved in describing multifarious features of brain signals

- The parameterised NMMs with the proposed parametrisation approach has been tested by fitting NMMs to the real EEG recordings. Preliminary investigation into the feasibility of using the fitted connectivity parameters for the classification purpose has also been carried out by comparing the difference in different sets of fitted connectivity parameters that were fitted to EEG recordings at different frequency bands.

1.4 Outline of the Thesis

The thesis is organized as follows:

Chapter 2 gives a basic introduction to the relevant aspects of brain functions, neuroscience, Brain-Computer-Interface (BCI) and state-of-the-art in BCI studies. The emphasis is then given to the general aspects of machine learning algorithms, in particular classification mechanisms and metaheuristics algorithms. Finally, an overview of Neural Mass Models (NMMs) as an appropriate approach to acquiring a deep insight into high-level brain rhythms, resulting in a better simulation of brain dynamics, is provided.

Chapter 3 presents the experimental protocol for Electroencephalogram (EEG) data acquisition. The data pre-processing procedure for enhancing the quality of the recorded signals prior

to the feature extraction has also been explained in detail. Finally, the feature extraction procedure is presented that prepares the feature vectors for the classification algorithms introduced in the remaining chapters.

Chapter 4 discusses the implementation of an Immune Inspired Multi-Objective Fuzzy Modelling approach for binary classification problems (IMOFM-C), which is originally designed for regression problems. In this chapter, IMOFM has been adapted for classification of multi-class SSVEP EEG data. IMOFM-C has been exploited to elicit a set of accurate and interpretable FRBSs which provide the basis for designing effective ensemble classifiers in Chapter 5. Furthermore, One-Against-One (OAO) decomposition-based approach is explained which is used to decompose the multi-class EEG data into six binary classification problems. Finally, The DDAG-Distance aggregation approach is proposed to aggregate the outcomes of IMOFM-C based binary classifiers. The proposed IMOFM-C classification mechanism has been tested on the SSVEP EEG data recorded from 4 human subjects and two benchmark datasets. Comparing to various baseline classification algorithms, such as adaptive neuro fuzzy inference system (ANFIS), artificial neural networks (ANNs),the CART decision tree algorithm, and Support Vector Machines (SVM), the proposed IMOFM-C based multi-class classification algorithm presents superior performance in terms of both accuracy and interpretability.

Chapter 5 presents a preference-based ensemble classification framework known as IMOFM-CP which is developed based on the IMOFM-C classification mechanism introduced in Chapter 4. IMOFM-CP is designed to promote the diversity among the elicited classification models within preferred and small Regions of Interest (ROIs) on the Pareto-front, IMOFM-CP better utilises the power of each individual classification model as a member of a committee classifier (i.e. ensemble classifier). IMOFM-CP is applied to both benchmark problems and the real-world multi-class EEG data sets. The results of IMOFM-CP are compared to those based on IMOFM-C. The results confirm the fact that aggregating a set of elicited classifiers leads to

more accurate classification performance than that of the individual classifier or, in the worst cases, equally good classification performance.

Chapter 6 presents a robust optimisation approach for parameterising a thalamic neural mass model that simulates brain signals such as those observed in EEG and local field potentials. The parameterisation approach proposed in this chapter is based on a single-objective Genetic Algorithms (GAs) incorporating more recent concepts in robust optimisation and a clustering algorithm. The proposed approach aims to tackle the stochastic nature of the extrinsic inputs. The clustering algorithm is used to identify several potential regions that corresponds to certain brain activities as reflected in the real EEG data. This systematic approach facilitates a more rigorous search of the wider parameter space under conditions of uncertainties introduced by stochastic extrinsic model inputs, as well as by minor variations in parameter values that simulate environmental and inter-individual differences in the model.

Finally, concluding remarks, new perspectives and future directions are given in **Chapter 7**.

Chapter 2

Brain-Computer-Interface and Neural Mass Modelling

The aspect of Brain-Computer-Interface (BCI) has constantly captured popular imagination. The feasibility of the idea in using brain signals rather than peripheral parts of the human body to interact with the external environment has fascinated generations of scientists, especially the neuroscientists alike in recent years (Kandel 2000, Schiff 2011). Deep understanding of complex functionalities of the human brain structure is a vital and often the first step towards having a successful BCI system. Although such an attempt has been pursued alongside the BCI technology since more than three decades ago (Georgopoulos et al. 1982), the understanding of human brain functionalities is still remained as one of the fast growing and open research areas.

Traditional ways of studying BCI often include acquiring appropriate brain signals and establishing classification models based on the acquired data (Mason and Birch 2003). However, in designing a successful BCI system with satisfactory classification accuracy, a fairly large number of participating subjects and experiments are required which put a significant constraint on studying BCI. Neural mass models can help to overcome this problem by modelling the Region of the interest (ROI) of the human brains. Furthermore, they allow us to reconstruct ROIs that are inaccessible to measurement. This de facto offers an effective way of simulating various brain signals across different subjects under varying testing environments (e.g. with different noise and interruptions).

This project represents one of the first attempts to bridge the gap between the traditional BCI approaches based on machine learning algorithms and neural mass models. In chis chapter, we first review the basics of neuroscience. The main principals underlying BCI are reviewed in Sections 2.1 to 2.4. Machine learning algorithms including linear and non-linear classification algorithms are discussed in Section 2.5. The review of the most established neural mass modelling approaches along with different parameterisation techniques are provided in Section 2.6. The review in this chapter is meant to be comprehensive but non-exclusive.

2.1 Brain Functions and Basic Neuroscience

Over the past two decades, the study of brain functions, involving information processing in a complex network of neurons, has attracted a significant attention among researchers in neuroscience (Rao 2013). Information processing underlying different brain functions, such as sensory, motor and cognitive functions, is carried out by a large group of interconnected neurons, which are the cells in charge of encoding, transmitting and integrating signals inside or outside the nervous system.

The typical structure of a neuron includes: 1) a cell body (the soma); 2) a tree-like structure with branches called dendrites connected to the soma; dendrites propagate the electrochemical stimulation received from other neural cells to the soma; and 3) a single branch called the axon that conveys the electrical impulses from the soma to other neurons (Rao 2013). Neurons communicate with each other through connections of the dendrite-axon pairs. Such connections are known as synapses through which signals (typically as chemical) are transmitted. Synapses can be excitatory or inhibitory. An excitatory synapse causes an increase in the local membrane potential of the postsynaptic cell which is called the excitatory postsynaptic potential (EPSP).

An inhibitory synapse causes a decrease in the local membrane potential of the postsynaptic cell which is known as the inhibitory postsynaptic potential (IPSP) (Rao 2013).

In the task of transmitting the information between neurons, changes occur in the electric potentials of neurons at rest. Such potential is called the resting membrane potential. When a neuron receives sufficiently strong inputs (the net effect of EPSP and IPSP) from other neurons, a short-lasting event occurs, leading to a rapid rise and fall in the membrane potential known as an action potential. The action potentials are the units of information transmissions, which occur at the interneurons levels and are characterised by certain amplitudes and durations. The transmitted information is believed to be encoded in the frequencies of action potentials known as the firing rate, as well as the timings of action potentials (Deco et al. 2008).

Regarding the brain organisation, the human nervous system is mainly divided into the central nervous system (CNS) and the peripheral nervous system (PNS). PNS consists of neurons connected to skeletal muscles, skin and sense organs, and neurons that control visceral functions such as the pumping of the heart and breathing. Another important part of the human brain is neocortex which is part of the cerebral cortex, arranged in six layers and is involved in higher functions such as sensory perception (sight, hearing, taste, smell and touch), generating motor commands (planning, controlling and execution of voluntary movements), spatial reasoning (ability to mentally manipulate 2-dimentioanl and 3-dimaentional figures), conscious thoughts and language. The main type of neuron in the cortex is excitatory pyramidal neurons and the other one is inhibitory interneurons (Rao 2013). The CNS consists of the brain and the spinal cord. The spinal cord conveys the motor-control signals from the brain down to the muscles and sensory feedback information from the muscles up to the brain. The top surface and the centre of the brain, where thalamus is located, is regarded as the relay station, conveying all the information from the sensory organs to the neocortex (Rao 2013).Thalamus consists of two main components: 1) the dorsal thalamus consisting of nuclei with relay cells projected to

neocortex; and 2) the ventral thalamus consisting of thalamic reticular nucleus (TRN). Another cellular component of the thalamus, apart from relay and reticular cells, is the interneurons (INs) (Sherman 2006). It is believed that thalamus, apart from being the relay station, is also involved in active feedback loops with the neocortex through many cortico-thalamic feedback connections known to exist between these two regions of the brain (Rao 2013).

Human brain shows neural oscillations which are rhythmic patterns generated through the interaction of neurons. These rhythmic patterns are classified into different frequency bands: 1) delta (1-4Hz) for the deep sleep state, 2) theta (4-8 Hz) for the deep meditation and light sleep, 3) alpha (8-13Hz) for wakeful relaxation with closed eyes, 4) beta (13-30 Hz) for waking consciousness and critical reasoning state, and 5) gamma (30-70 Hz) for burst of insight and high level of information processing (Buzsáki 2006). These oscillatory patterns serve as important indicators (through different amplitude and rhythmic patterns) to the onset of certain pathological changes and intentions, such as in detecting neurological disorders including the seizure, epilepsy, and Alzheimer disease (Parvez and Paul 2014 , Al-Jumeily et al. 2015, Bhattacharya, Coyle, and Maguire 2011) , and in classifying different intentions in BCI (Akilandeswari and Nasira 2014).

When a group of neurons generate neural oscillations, they can be measured through different techniques (Sen Purkayastha, Jain, and Sardana 2014), among which Electroencephalogram (EEG) is one of the most popular ones, receiving much attention since its inception (J. R. Wolpaw and Wolpaw 2012) and will be expanded in more details in Section 2.3. This provides direct and non-invasive access to the reference signals at ROIs and sheds light on more accurate neural mass modelling.

2.2 Brain-Computer Interface (BCI)

Brain's Central Nervous System (CNS) interacts with the external world by generating neuro-muscular or hormonal outputs. BCI is a computational system that measures activities of the CNS and converts such activities into artificial outputs. The artificial outputs replace, enhance or improve natural CNS outputs. The artificial outputs ultimately change the interaction between the CNS and its external environment (J. R. Wolpaw and Wolpaw 2012). Therefore, BCI represents one of the significant and latest solutions that extends human's controlling capabilities directly by the means of brain signals. The main steps involved in order to create a communication link between the brain and external world include: 1) recording brain signals with either invasive or non-invasive recording techniques (Gert Pfurtscheller, Brendan Allison 2010); 2) signal pre-processing and feature extraction; and 3) classifying the feature vectors into appropriate commands for interacting with the outside world (Rao 2013, Wolpaw and Wolpaw 2012, Shih, Krusienski, and Wolpaw 2012). BCI has been used for different types of applications including medical and non-medical ones (Abdulkader, Atia, and Mostafa 2015).

In medical applications, BCI tries to overcome functional deficits. An example of this is to design a vision based non-invasive BCI for grasping an object using a robotic arm. In this case, a user wearing an EEG headset chooses an object from a scene image displaying in a visual stimuli grid. After going through the main steps mentioned above, the command will be generated and sent to the robotic arm, and the action will be performed. This kind of the BCI system can help patients suffering from certain forms of disability which makes them unable to perform their daily tasks independently or without the aid of a human carer. It is particularly useful for patients suffering from a brainstorm stroke or amyotrophic lateral sclerosis, who eventually become completely locked-in and are unable to move any muscles in the body (Pathirage et al. 2013).

Another example for using BCI in medical applications is for treating a number of cognitive neurological disorders such as predicting and controlling seizure. In these works, real-time detection and prediction algorithms (Osorio, Frei, and Wilkinson 1998), responsive simulation (Morrell 2011) and closed-loop control systems for the seizure detection are employed (Xin-Chun Zhang 2013). They are incorporated into BCI to monitor the onset of the seizure. Upon its detection, appropriate drugs can be delivered to supress the seizure before it spreads to other parts of the brain (Qi et al. 2014, Luijtelaar 2013, Rao 2013).

There has been a significant rise in the number of non-medical applications. One notable example is to monitor alertness of humans during the performance of critical tasks, such as driving (Mardi, Ashtiani, and Mikaili 2011). Having the technology to monitor the vigilance of drivers can prevent catastrophic accidents which are caused due to tiredness or drowsiness driving. It has seen many EEG-based BCI studies applied to monitoring brain signal transitions from an alert state to a state indicating lack of alertness (Wang and Li 2012, Rao 2013).

2.3 Measuring Brain Activities (Non-Invasive vs. Invasive Techniques)

There are two classes of methods in measuring brain activities, namely invasive and non-invasive techniques. Invasive techniques take measurements from individual neurons of the brain. Therefore, some forms of surgery to remove a part of the skull are required. An electrode is then placed in the brain to replace the removed part of the skull. As this method involves a brain surgery, it can cause pain during the surgery and require a recovery process. It also runs the risk of infections. However, the major benefit of invasive techniques over non-invasive ones lies in its capability to record action potentials in the millisecond timescale. Intracellular

recording technique is an example of invasive technique which measures the voltage and current across the membrane of the neurons. In this technique, a tip of a fine microelectrode must be inserted inside the cell in order to record the membrane potential. A microelectrode is a simple wire with a sharp pin or other electrical conductor which are used to make the contact with the brain tissue (Rao 2013). Non-invasive techniques measure indirect correlations of neural activity such as the blood flow which occurs at hundreds of milliseconds. Non-invasive techniques do not involve a brain surgery. Among many non-invasive techniques, EEG is one of the most popular non-invasive measuring techniques which applies the electrodes onto the scalp for the recording purpose. EEG records the brain's spontaneous electrical activity over a period of time from the multiple electrodes placed on the scalp (F.H.Lopes Da Silva 1982).

To conduct EEG recording experiments, an internationally recognised method, known as the 10-20 international System, is developed (Rao 2013). 10-20 international system defines the locations of the scalp electrodes in the context of an EEG test (as shown in Figure 2.1) (Acharya et al. 2016).

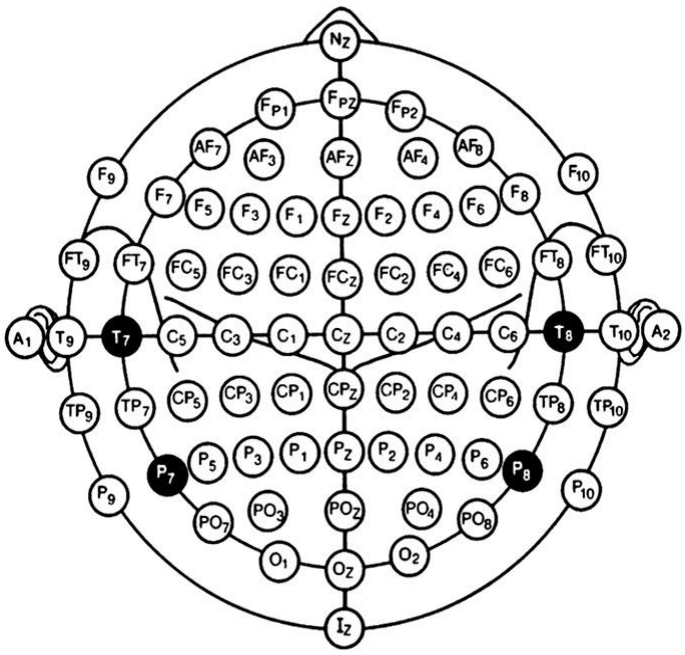

Figure 2.1 10-20 International System (Acharya et al. 2016).

Here, “10” and “20” correspond to the distances between the adjacent electrodes which are either 10% or 20% of the total front-back or right-left distances of the skull. The letters, F, T, C, P and O, stand for Frontal, Temporal, Central, Parietal and Occipital lobes respectively (Acharya et al. 2016).

2.4 EEG Brain Signals

2.4.1 Event Related Potentials (ERPs)

Event Related Potentials (ERPs) are very small voltages generated in the brain structures in response to stimuli (Sur and Sinha 2009). It is believed that sensory stimuli events can cause time-locked changes or ERPs which are seen in the activities of neural population. One of the main techniques for detecting event-related potentials is averaging as it is believed that ERPs activities have fixed time delays to the stimuli, while the ongoing EEG activities behaves like additive noise; moreover, the averaging technique enhance the signal to noise ratio. The P300 wave is one of such ERPs and is usually elicited using the oddball paradigm. The oddball paradigm is an experimental designed platform used for ERP research, in which a sequence of repetitive audio or visual stimuli are represented and infrequently interrupted by a deviant stimulus (J. Wolpaw and Wolpaw 2012). An example of such an experimental platform is the P300 speller (Guan, Thulasidas, and Wu 2004). The presence of stimulus in an oddball paradigm will produce a positive peak as the response with a latency between 250 to 750 milliseconds (Comerchero and Polich 1999). The electrodes that P300 are usually recorded from comply with the 10-20 international system. The amplitude of the P300 wave is in the range of 2 to 5 μV. Considering its low amplitude with respect to the background activities of the brain, EEG is averaged over multiple trials to enhance the amplitude of the P300 signal and for a better

detection of P300 (Amiri et al. 2013). Figure 2.2 shows an example of the average P300 response versus the average background EEG (Non- P300). As P300 can be detected through non-invasive and portable devices, P300-based BCIs have received a growing interest over the past two decades as an inexpensive and reliable platform (J. Wolpaw and Wolpaw 2012).

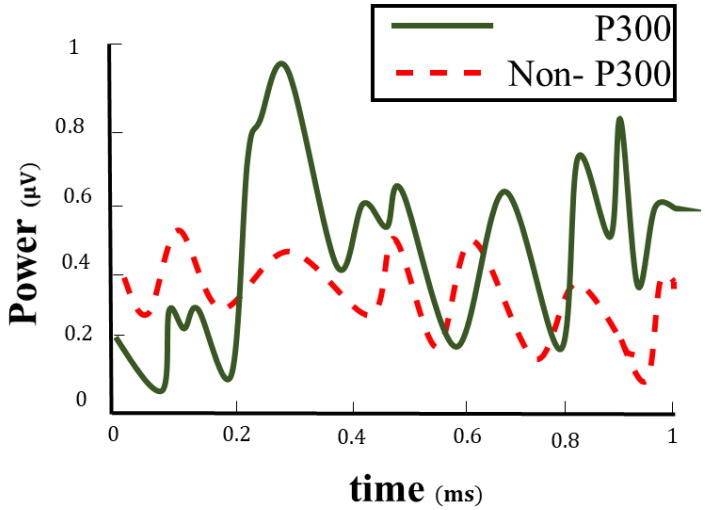

Figure 2.2 An example of the average P300 response versus the average background EEG.

2.4.2 Induced Activities (ERD/ERS)

Unlike ERP activities, induced activities are triggered by visual stimuli in which there might be time locked changes but not phased-locked. In another words, induced activities make frequency changes in ongoing EEG activity which can lead to decrease or increase of amplitude in a specific frequency. This event can be interpreted as increasing or decreasing of the synchrony of the underlying neural populations; the former called Event-Related-Synchronisation (ERS) and the latter is known as Event-Related-Desynchronisation (ERD) (Pfurtscheller and Lopes da Silva 1999). As it is shown in Figure 2.3, ERD corresponds to decrease in power, while ERS shows an increase in signal power (Lemm, Müller, and Curio 2009)

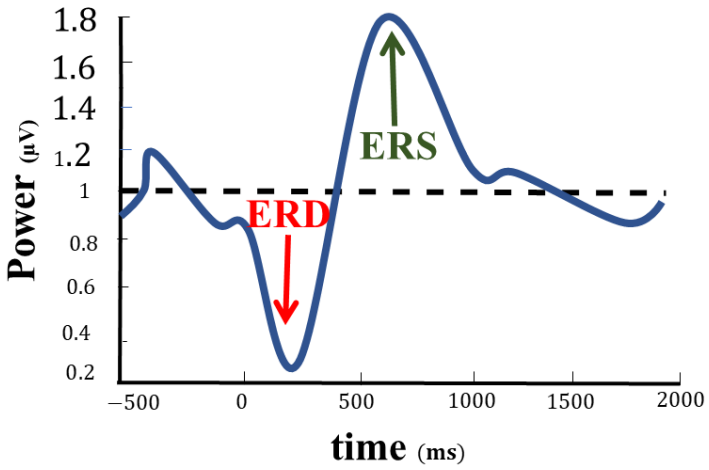

Figure 2.3 An example of Event Related Synchronisation (ERS) and Event Related-Desynchronisation (ERD) distinguished from constant baseline level (black dashed line).

2.4.3 Steady State Visual Evoked Potential (SSVEP)

Steady-State-Visual Evoked Potentials (SSVEPs) are stable oscillations in voltage that are evoked by sudden visual stimuli such as a light flash or appearance of an image. The successive stimulus representations evoke similar responses and the overlap of these responses produces a steady-state oscillation (Middendorf et al. 2000). SSVEPs can be measured by traditional averaging method or frequency analysis in which peaks at the frequency of stimulation and also other harmonic frequencies appear. SSVEP has been used as one of the main brain signals for designing BCIs (Middendorf et al. 2000, Cheng et al. 2002, Kelly et al. 2005) as it demands less attention and is less sensitive to artefacts as long as the frequencies of the artefacts are not overlapping with the stimulus frequencies (Leow, Ibrahim, and Moghavvemi 2007). In a standard SSVEP based BCI system, a visual field is provided which represents concurrent visual stimuli for a user. Each stimulus has a specific frequency which makes it different from other stimuli. User is asked to fixate on a specific stimulus depending on a desired BCI output (e.g. type a letter, control a device, etc.) that each stimulus represents. The peak will be seen in the

frequency spectrum which matches the rate of the stimulus that the user was fixating on and the desired BCI output will be produced.

Figure 2.4 illustrates an example of SSVEP response around frequencies of 14 Hz (a) and 21 Hz (b), which are evoked by 14Hz and 21Hz stimuli respectively.

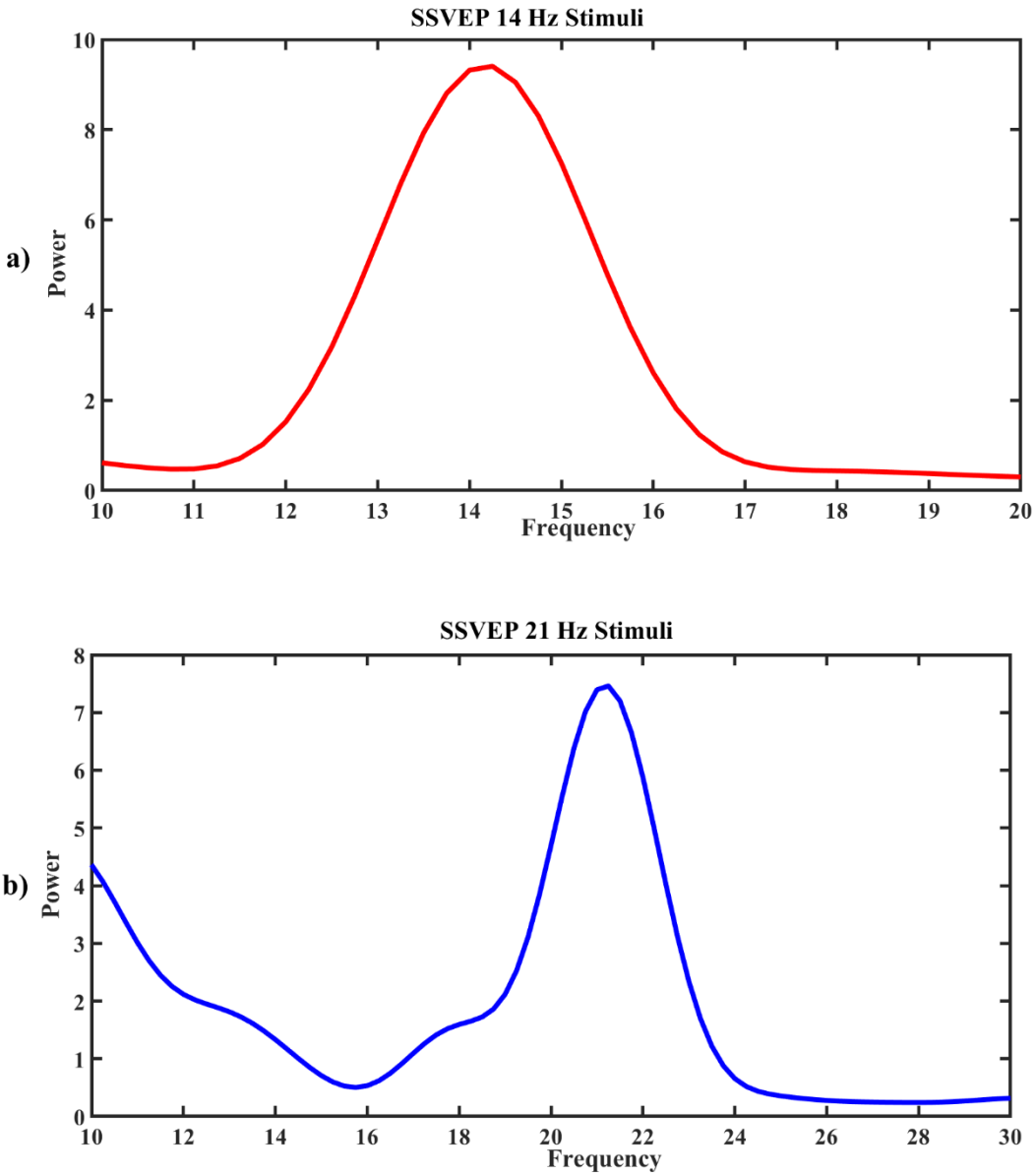

Figure 2.4 An example of SSVEP signals a) 14 Hz Stimuli and b) 21 Hz stimuli

2.5 Machine Learning for BCI

The field of machine learning plays a vital role in the development of BCI. It encompasses techniques that can learn to map neural activities into appropriately generated commands. Generally speaking, algorithms in machine learning are divided into two categories: unsupervised and supervised learning (Htike and Khalifa 2010). In unsupervised learning, data is not labelled, and the algorithm is based on exploring and extracting hidden statistical structure by using probability density models or similarity matrices (Xu and Wunsch 2005). Clustering analysis falls into this category. In supervised learning, training data are provided which consists of a set of inputs (i.e. features) and corresponding outputs (i.e. labels). The aim is to learn the underlying relationship between the features and labels from the training data. When a new concept appears, the algorithm will associate certain features with a label. One of the most common supervised learning techniques is classification algorithms. In the following, EEG data pre-processing and feature extraction techniques will be first reviewed, followed by unsupervised learning and supervised learning.

2.5.1 EEG Data Pre-processing

EEG signal pre-processing needs to be conducted so that signals will be enhanced by eliminating known interference, such as artefacts and noise, while their spectral, temporal or spatial characteristics will be intensified. EEGLAB is an example of open-access toolbox which is commonly used to analyse EEG data (Delorme and Makeig 2004). The pre-processing procedure includes four main steps which have been briefly explained below.

1. **Data Decimation:** Decimation is to eliminate samples in a periodic fashion (J. Wolpaw and Wolpaw 2012) When the signal is digitised at higher rate than required, for more efficient processing and less required data storage, it can be beneficial to decimate the sampled signal to a minimum and effective sampling rate.

2. **Filtering:** EEG signals of interest are normally in a specified range of frequencies. Therefore, eliminating the influence of the frequencies that lay outside the range of frequencies of interest is necessary. Toward this, temporal filters, including Finite Impulse Response (FIR), Infinite Impulse Response (IFR), and Discrete Fourier Transform (DFT) can be used to remove undesired effects and isolate relevant brain activities. It is worth mentioning that the width of the filter (i.e. the frequency range) should be selected carefully to prevent unnecessary loss of information. For example, if the purpose is to isolate alpha-band brain activities a bandpass filter of 8-13 Hz will be applied. In the case of ERPs, such as P300, as the aim is to preserve the characteristic low-frequency information of the response, the high-pass cut off of 0.1-0.5 Hz and a low-pass cut off 5-40 Hz are applied. In the case of SSVEP, the high-pass cut off of 0.5 Hz and the low-pass cut off of 100 Hz are used.

3. **Segmentation:** EEG signals are commonly segmented into consecutive sample blocks prior to the feature extraction process. The feature vectors are generated from the samples within each individual sample block.

4. **Artefact reduction:** In EEG recordings, there are various types of artefacts which arise from biological sources such as muscle activities (using electromyographic (EMG)), eye

movements (using electrooculographic (EOG)), and heart-beat activities (using electrocardiographic (ECG)) (J. Wolpaw and Wolpaw 2012). Muscle activities are one the most significant artefact as it is very difficult to be removed or fully realised. To reduce artefact related to muscle activities, spatial filtering techniques can again be employed. For the eye-blink artefact, as it usually appears at the low-frequency, bandpass filters can be applied (J. Wolpaw and Wolpaw 2012).

2.5.2 Feature Extraction Techniques

A feature is a distinctive measurement extracted from a segment of a signal. Features are used to represent relevant characteristics of a signal. There are different feature extraction techniques in time and frequency domains. As feature extraction and machine learning techniques need to be synergized, the choice of feature extraction techniques will affect learning methods and vice-versa. In the following, several feature extraction methods relevant to this project are briefly reviewed.

1. Frequency-domain approaches. Many BCI applications rely on the features and the changes that can be tracked in the frequency-domain.

    a. Band power is one of the most straight-forward methods for tracking amplitude modulation at a particular frequency; amplitude modulation is the variation of amplitude which follows a specific pattern. First, the range of the frequency of interest is identified and isolated by using bandpass filters. Then, the power spectrum in a particular frequency band is computed and taken as the spectral feature for further analysis (Wolpaw and Wolpaw 2012, Rao 2013).

b. Fast-Fourier Transform (FFT) is commonly used to transfer a time-varying signal into the frequency domain in order to extract the frequency features of the signal.

2. Time-domain approaches.

   a. Peak-Picking and Integration are both the basic feature extraction methods. Peak-picking utilises the minimum or maximum values of a signal in a specific time block. These values serve as the features of that time block. Furthermore, signals can be averaged or integrated over all or part of the time block, generating features for that time block.

   b. Correlation and template-matching compute similarity or correlation of a signal to a predefined template. For a given response, the output will be high in the case of close resemblance of the segments to the template, and low if the segments differ from the template (J. Wolpaw and Wolpaw 2012).

3. Time-frequency approach.

   Wavelet Transform (WT) approach is a time function of simple and fixed blocks known as wavelets. The building blocks of WT are a family of functions which are derived from a single generating function called mother wavelet which involves translation and dilation operations. Dilation is a scaling function which compress or stretches the mother wavelet. Translation shifts the mother wavelet along the time axis (Cvetkovic, Übeyli, and Cosic 2008). WT is divided into two categories, namely continuous WT (CWT) and discrete WT (DWT).

   In CWT, due to the continuous varying scaling and translation parameters, the wavelet coefficients are required to be computed for every single scale. Therefore, it results in huge computational time and vast amount of data (Subha et al. 2010).

DWT also outperforms the aforementioned time/frequency-domain approaches due to the following main reasons. (1) EEG signals are non-stationary and possess transient patterns. Traditional methods, such as FFT, is not sufficient as they only extract non-transient information in the frequency domain. Furthermore, the time-domain information is not readily seen from the transformed Fourier coefficients ((Adeli, Zhou, and Dadmehr 2003). Whilst, DWT is a powerful time-frequency method in capturing transient information such as sudden short-duration signal changes which are common in time-varying biomedical signals (Cvetkovic, Übeyli, and Cosic 2008). (2) FFT is not a sufficient approach to extract frequency-domain information as it suffers from low-frequency resolution, as well as spectral leakage due to its windowing function. DWT can provide time-frequency localisation with a flexible window size. A wider window size is used to produce good frequency localisation at the lower frequency and a narrower window size is used for better time-localisation at the higher frequency (Mallat 1989, Jahankhani, Kodogiannis, and Revett 2006).

2.5.3 Unsupervised Learning and Clustering

Clustering techniques are based on unsupervised learning. They do not use categorical labels based on prior knowledge. The absence of the prior information distinguishes clustering techniques from supervised learning which requires the prior knowledge (i.e. labels) (Jain 2010). Clustering groups data instances into subsets so that similar data instances are regarded as within the same class and different data instances belong to different classes (Fayyad, Piatetsky-Shapiro, and Smyth 1996). Clustering analysis involves measurement process in order to determine whether two observers are similar or dissimilar. Two types of measurements, namely distance measures (e.g. the distance between two observers) and

similarity measures (e.g. the magnitude of their similarity to each other) are widely adopted (Späth 1980). Clustering algorithms are mainly classified into two groups.

1. Exclusive clustering. Data are grouped in an exclusive way, into partitions, so that a certain data point belongs to only one particular cluster. K-means clustering is one of the main examples of this type (Jain 2010). K-means algorithm is one of the simplest unsupervised clustering algorithms. In the first stage of this algorithm, K random centres are chosen for each cluster. The next stage is to assign each data points to the nearest centres based on the distance measure. Euclidean distance is used to determine the distance between each data point and their cluster centres. When all data points are grouped into different clusters. The positions of the k centres are recalculated. This iterative process is carried out until it minimises the squared error function (Na, Xumin, and Yong 2010). K-means algorithm does not necessarily converge to the global minimum solution. It is also very sensitive to the initial cluster centres. Another notable algorithm falls into this realm is K-Medoids algorithm (Kaufman and Rousseeuw 1987) which is more robust compared to the K-Means. In K-medoids clustering, a data point which is the most centrally located in a cluster known as the medoid, is used. Instead of minimising the squared Euclidean distances, it minimises the sum of dissimilarities between each data point and its corresponding cluster centre.

2. Hierarchical clustering algorithms. There are two main types (Jain 2010). Agglomerative clustering starts with each data point in a particular cluster and gradually merge two most similar clusters. On the contrary, divisive clustering starts with all data points in one cluster. At each step, clusters are successively spitted into smaller clusters according to some dissimilarity measures.

2.5.4 Supervised Learning and Classification

Classification is a supervised algorithm that maps (classifies) data into one of the predefined classes (Fayyad, Piatetsky-Shapiro, and Smyth 1996). In other words, classification algorithm requires labelled training data, consisting of known features and their corresponding labels (classes). When a new feature appears, the classifier assigns one of the labels to it (Rao 2013). In the field of BCI, different patterns of brain activities are generated and BCI aims to translate these brain activities into different control commands.

2.5.4.1 Classification Approaches

Linear approaches. Linear approaches assume a linear separability of data points. Linear classifiers are generally efficient as they have less parameters to tune. However, in the presence of extensive noise and outliers, the linear classifiers will fail (Gandhi 2014). They also suffer from the 'curse-of-dimensionality', which occurs when the amount of sampled data required for describing different classes increases significantly with the increase in dimensionality of feature vectors (Müller, Anderson, and Birch 2003). In the following, an example of the linear classifiers is explained.

Linear Discriminant Analysis (LDA): LDA is by far the most widely adopted classifier for BCI (Ianez et al. 2010) LDA has a very low computational requirement and is simple to implement, making it a suitable candidate for online BCI application. LDA classifiers project a P dimensional input by establishing a hyperplane which separates the input space into two half spaces. The decision boundary in LDA is provided by the hyperplane as defined in Eq. 2.1:

$$g(x) = w^T x + w_0 = 0 \qquad (2.1)$$

The boundary between two classes is characterised by the weight vector w and the threshold W_0 ,W and W_0 are elicited from the training data. The classification for the new input x is computed using Eq. 2.2.

$$y = sign(w^T x + w_0) \tag{2.2}$$

The positive sign in Eq. 2.2 determines that the input is belonging to the first class. While the negative sign shows that the input is in the second class. The LDA classifier aims to maximise the distance between two classes while minimising the interclass variance (Lotte et al. 2007, Cherkassky 2005). However, the prerequisite of linearity is its drawback as it cannot handle the complex nonlinear problems. To overcome this issue, Support vector machine (SVM) is often employed, which transfers the non-linear data into a high dimensional and linear separable space.

Nonlinear approaches. Linear methods are not always effective, particularly for large complex problems with nonlinearity. To address these issues, nonlinear classifiers are chosen. In this section, three nonlinear classifiers are introduced including Artificial Neural Networks, Support Vector Machines and Fuzzy Inference Systems.

Artificial Neural Networks (ANNs). ANNs are inspired by the structure of human brain which compromises individual neurons, connection weights, a summing node which sums all the inputs received by each neuron, and a subsequent activation function which generates the output (J. Wolpaw and Wolpaw 2012). A neural network presents a mapping from the input space to the output space. The mappings at each neuron are performed in parallel. The information processing in each neuron is nonlinear. Therefore, the overall mapping is characterised as a nonlinear process. One of the most important properties of ANNs is their ability to learn from the training set in order to generate a certain output given a new input. In the training process, every example has an input pattern associated with a desired output

pattern. To train the network, an example from the training set is provided to the network and the generated output is observed. If the generated output is different from the actual output, the weight connections are modified to minimise the difference between the actual output and generated outputs (Forslund 2003). The structure of an ANN consisting of a single unit neuron is illustrated in Figure 2.5 (J. Wolpaw and Wolpaw 2012), in which the inputs on the left are weighted and summed. If the sum reaches the threshold as defined in the activation function, an output will be produced.

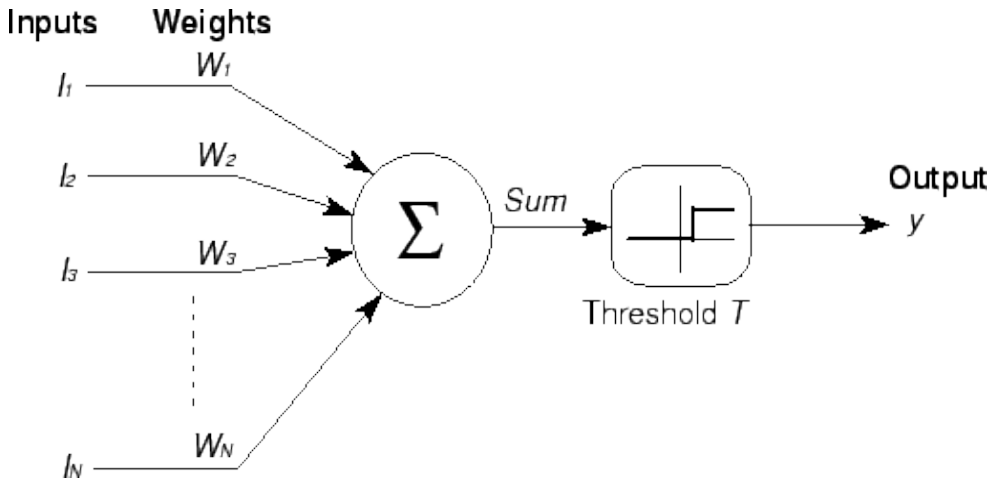

Figure 2.5 The structure of single neuron neural networks (J. Wolpaw and Wolpaw 2012).

Theoretically, ANNs are capable of approximating any linear/nonlinear functions provided sufficient neurons and hidden layers are included (Carpenter and Grossberg 1991). However, the increase in the number of neurons increases the complexity of ANNs. The most well-known form of ANNs is the multi-layer perceptron (MLP) (Carpenter and Grossberg 1991)which is an example of multi-layer feedforward neural ANN as shown in Figure 2.6 (J. Wolpaw and Wolpaw 2012). In this example, an ANN compromises of three inputs representing different dimensions of feature vectors, several hidden layers in between the inputs and outputs. The two outputs represent two possible BCI commands.

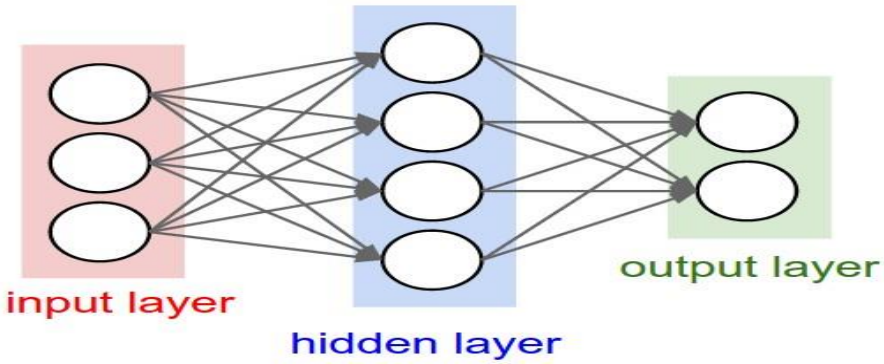

Figure 2.6 An example of multi-layer Perceptron neural network (J. Wolpaw and Wolpaw 2012)

In the mapping process, the inputs are fed to the network and the output is produced. As every individual weight needs training, leading to a large amount of time for the training process (J. Wolpaw and Wolpaw 2012).

Support Vector Machines (SVMs): SVMs can be applied to both linear separable and non-separable problems. SVM classifiers aim to find the optimal hyperplane that is maximising the margin between the two classes. In the simplest case, linear SVM classification is trained with the data which are linearly separable and generates a hyperplane that defines the border between two different classes. In the non-separable case, a nonlinear mapping of data to a sufficiently high dimensional space is conducted first so that the two classes are linearly separable after the mapping. Then, a classic linear SVM can be applied for such a non-separable problem (Rao 2013, Hofmann 2006, Lotte et al. 2007). However, SVM suffers from the following issues: 1) in the nonlinear case, having the accurate mapping is critical; otherwise, the poor mapping will significantly affects the classification accuracy; 2) mapping data into a high-dimensional space can lead to a data loss; and 3) SVM is a black-box classifier, meaning the human knowledge cannot be easily incorporated.

Fuzzy Inference Systems (FISs)and Fuzzy Rule Based Systems (FRBSs): In real world, most of the systems show not only non-linearity but also uncertain behaviours. Fuzzy systems can overcome these issues by incorporating human knowledge into the sensory measurements and mathematical models (Jun 2010).To achieve this, fuzzy logic is introduced via the concept of the fuzzy set (Jun 2010) providing a set of qualitative rules that can express nonlinear input/output relationships (Chiu 1994). A formal definition of the fuzzy set is given by Zadeh (1956) (L.A. Zadeh 1965) as below.

"Let X be a space of points (objects), with a generic element of X denoted by x. Thus, $X = \{x\}$. Then a fuzzy set (class) A in X is characterised by a membership (characteristic) function $f_A(x)$ which associates with each point in X a real number in the interval [0,1], *with the value of $f_A(x)$ at x representing the grade of membership of x in A."*

A membership function is a function that assigns a membership value to each point in the input space which determines whether the element belongs to a particular fuzzy set and to what degree (degree of certainty) (Jun 2010). Membership value of a classical set can only be a discrete value, i.e. either 0 or 1, which reflects the black or white situations. While, fuzzy membership blurs this discrete division and leads to extra smoothness (Jun 2010).

Fuzzy Inference is the process of formulating the mapping from a given input to an output, where the mapping provides the basis from which the decisions can be made. Fuzzy inference systems are also known as Fuzzy Rule-Based Systems (FRBSs). FRBSs are based on classical rule-based systems (Riza et al. 2015). A FRBS is usually represented by rules of the following form "$If\ A\ then\ B$", where A and B are fuzzy sets and they are called antecedents and consequents of the rule, respectively. The general process of fuzzy inference system (FRBSs) is illustrated in Figure 2.7 (Jang 1993).

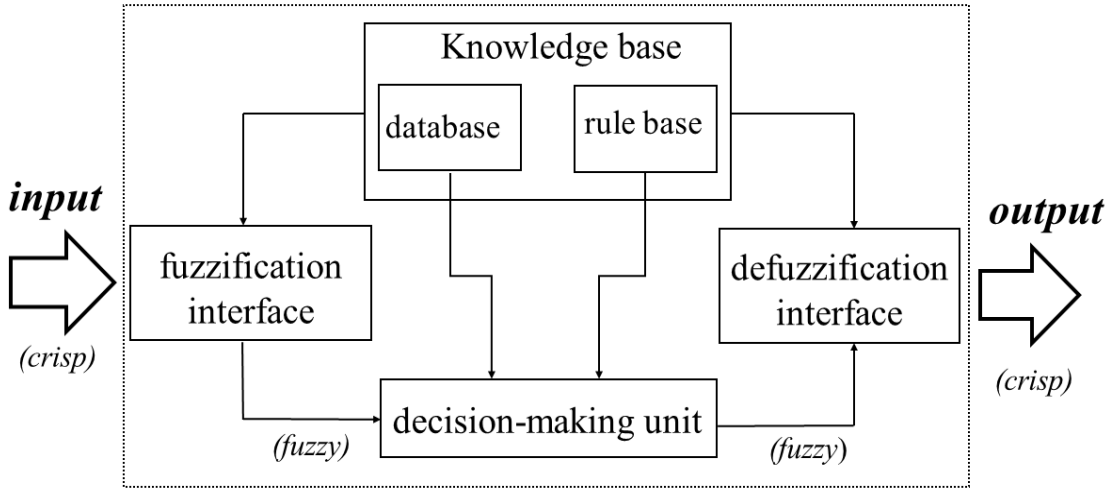

Figure 2.7 Fuzzy Inference System (FRBS)

As it shown in Figure 2.4 the fuzzy inference includes: 1) a 'rule base' containing set of fuzzy-if-then rules; 2) a 'database' which defines the membership functions of the fuzzy sets; and 3) a 'decision-making' stage which performs the inference operation based on the rules and two interfaces which perform fuzzification and defuzzification. In the fuzzification step, it takes the crisp input and converts it into a fuzzy input. In the defuzzification step, a fuzzy set, representing the inferenced result, is transformed into a crisp value (Jun 2010).

The two most popular types of FRBSs are the Mamdani-type (Mamdani 1974), and Sugeno-type (Takagi and Sugeno 1985). Mamadani-type is based on Zadeh's fuzzy algorithms for complex systems (Lotfi A. Zadeh 1973). These two types are different in the way that outputs are determined. The consequent part of the Mamdani-type is a fuzzy set while the consequent part of the Sugeno-type is a set of functions with the argument that are the linguistic variables of the antecedent part (Jun 2010).

In the work by Lotte (Fabien et al. 2007), FISs have been used for classification of EEG signals. They initialise the rule based of the classifier through clustering. At the time of applying the

trained classifiers, they extracted the band power features of the EEG data recorded from the particular electrodes and based on the feature, two fuzzy rules were extracted. It is stated in this work that, the FIS classifier outperforms the linear classifier and is as accurate as the SVM classifier. Further advantageous, such as being interpretable and extendable, make it suitable for a real-time BCI application.

2.5.4.2 Ensemble-based Classification

Ensemble learning is an umbrella term for methods that combine multiple classifiers to make a decision in supervised machine learning tasks. An ensemble classifier can be of any type of machine learning algorithms e.g. ANN, SVM, FIS, etc. The main premise of ensemble learning is that by combining multiple classifiers, the errors of a single classifier will likely be compensated by other classifiers, hence, the overall prediction performance of the ensemble would be better than that of a single classifier (Sagi and Rokach 2018).

In recent years, ensemble learning is considered as the state-of-the art approach for solving a large amount of machine learning challenges as shown in an extensive comparison of 179 classifiers from 17 families using 121 datasets from UCL (Fernández-Delgado et al. 2014).

Ensemble methods often improve the overall predictive performance due to several main reasons (Dietterichl 2002, Sagi and Rokach 2018):

- Overfitting avoidance: In the case of having a small amount of training data, a single classifier is prone to predict all of the training data perfectly correct while its prediction for the unseen data can be very poor. Therefore, using an ensemble of classifiers and averaging the prediction of each classifier reduce the risk of poor prediction for the unseen data and hence improve the overall prediction accuracy.

- Computational Advantage: a single classifier that explores the search space may get stuck in local minima; however, using an ensemble of classifiers would reduce the risk of trapping in the local minima.

- Representation: The search space for a single classifier can be limited, therefore, the optimal prediction for an unseen data might not be achieved. However, combining multiple classifiers can explore more search space and hence, a better prediction can be obtained for unseen data.

- Class imbalance avoidance: In many classification problems, there are cases where one class has more samples comparing to the other classes. In such cases, the classifier can bias the prediction in favour of the majority class and ignoring the minority classes (Japkowicz and Stephen 2002). In a work by (Galar et al. 2011), it showed combining the random under sampling techniques with ensemble approaches e.g. bagging or boosting may significantly improve the prediction accuracy for problem with class imbalance (Sagi and Rokach 2018).

The key principals of designing an ensemble of classifiers are:

1. Diversity: An ensemble of classifiers can perform superior than a single classifiers by use of various "inductive biases" (Deng et al. 2013). Therefore, the participating classifiers should be sufficiently diverse to improve the overall prediction accuracy.

2. Accuracy: To design a diverse and accurate ensemble of classifiers, the accuracy of each individual classifiers should be as high as possible.

There are several main approaches for generating diverse and accurate ensemble models:

1. Classifiers combined with evolutionary algorithms: Combining machine learning based classifiers with evolutionary algorithms (EAs) is one of the main approaches in

literature to form a diverse yet accurate ensemble of classifiers. The evolutionary process of this approach will generate a population of classifiers which are different from each other and hence contribute to their diversity.

Evolutionary algorithms are generic population based-metaheuristic optimisation algorithms. Genetic Algorithm (GA) is the most popular type of evolutionary algorithms which are used in optimisation problems. There are two main types of optimisation, i.e. Single-Objective Optimisation (SOP) and Multi-Objective Optimisation (MOP). SOP is the basis of all types of optimisation. Many real-world decision-making problems need to achieve several goals simultaneously, e.g. maximising profit, minimising risk, etc. The main goal of SOP is to find an optimal solution which corresponds to the minimum or maximum of a single objective function that converts all different objectives into one. This type of optimisation is useful as a tool to provide insights into the nature of the problem for decision makers (Savic 2002). However, many real-world problems are inherently of a multi-objective nature with often conflicting goals. Therefore, in multi-objective optimisation with conflicting objectives, there is no single optimal solution. Instead, the interaction between different objectives results in a set of trade-off solutions, known as non-dominated or Pareto-optimal solutions (Savic 2002). The Pareto optimal solutions are the points exist along the Pareto front. The concept of Pareto front or set of optimal solutions in the space of objective functions in MOP problems represents a set of solutions that are non-dominated to each other but superior to the rest of the solutions in the search space. In other words, a solution is called non-dominated or a Pareto-optimal solution if none of the objective functions can be improved in values without degrading some of the other objective values.

An example of combining the machine learning-based classification algorithms with EAs is a work by Xin Yao's work in 1998, in which a simultaneous evolution of ANN architectures and weights was introduced. Four linear combination methods were investigated for combining different individuals in the last generation into an ensemble (Xin Yao and Yong Liu 1998). Later, in 2004, Arjun Chandra and Xin Yao, introduced the DIVerse and ACcurate Ensemble learning algorithm (DIVACE), which formulates the ensemble learning of ANNs as a multi-objective problem explicitly (Chandra and Yao 2004). In DIVACE, the two objectives against which the performance of the ensemble is optimised are accuracy and diversity.

2. Bagging: Another approach for creating a diverse set of classifiers is through data resampling. Breiman's bagging (short for bootstrap aggregating) is one of the earliest ensemble algorithms designed based on data resampling (Breiman 1996). In Bagging, diversity can be achieved through randomly generated different subsets of training data with replacements from the entire training data. Then, each generated subset of training data is used to train a different classifier of the same type. Finally, generated individual classifiers will be combined using a majority vote of their decisions (Polikar 2006). Bagging is commonly used when the available training data has a limited size. In this scenario, relatively large portions of the samples (e.g. 75% or higher) are drawn into each subset to make sure each subset has adequate samples of training data. Therefore, there is a significant overlap between individual training subsets and many of the same instances appear in most subsets at least once or multiple times. There are different variations of the bagging algorithm such as Random Forest and Pasting Small Votes (Polikar 2006).

3. Boosting: Boosting is another ensemble approach designed based on the data resampling strategy (Schapire 1990). The basic idea of this algorithm is to improve the

performance of the weak classifier by repeatedly running it on the various distributed training data. Therefore, the classifiers generated by the weak learners are combined through majority voting and form a single strong classifier which can achieve a higher accuracy than the best classifier in the ensemble (Rokach 2010). AdaBoost is one of the variants of the boosting algorithm proposed by Freund and schapire (Freund and Schapire 1997). The main idea of AdaBoost algorithm is to be more focused on the patterns that are harder to classify. In this approach, initially all the patterns are assigned the same weight or will be given the same amount of focus. However, gradually throughout an iterative process, the given weights to the misclassified patterns will be increased, while the weights for the correctly classified patterns will be decreased. As a result, the weak learner is more focused on the difficult patterns of the training set by performing additional iterations, generating more classifiers. Finally, the outputs of the generated individual classifiers will be combined through the weighted majority voting to yell the final decision on each data instances (Rokach 2010).

2.6 Neural Mass Modelling

Neural field modelling is one of the approaches that facilitate studying the brain and modelling neural signals, in which the oscillations can be explained and described using rigorous mathematical modelling tools (Schiff 2011). However, understanding the rhythmic oscillations is not a trivial task due to the presence of intrinsic membrane properties, non-linearity and the interaction of synaptic conductance (Rao 2013).Therefore, choosing an appropriate approach for modelling the neural populations plays an important role in investigating neural oscillations. There are two types of modelling approaches: detailed modelling (microscopic) and reduced

one (macroscopic). In detailed modelling, the aim is to capture the complexity of each individual neurons and synaptic connectivity. Several principles have been proposed for the detailed modelling: 1) using models of single neurons that contain dendrites, soma and part of the axon; and 2) in order to reflect the actual size of the network of neurons, there is a need to use enough neurons and different types of interneurons, leading to an increased model complexity. The reduced modelling only involves a smaller number of neurons as it aims to investigate the vital features for a particular dynamic behaviour. Neural mass models are based on the reduced modelling and it has been widely studied and used compared to detailed modelling. The application areas of neural mass models have been broadened over the past decades, such as modelling alpha rhythms for medical applications (Lopes da Silva et al. 1974), and studying pattern changes due to the variation of the synaptic connectivity between different population cells of the human brain (Basabdatta Sen Bhattacharya, Coyle, and Maguire 2011b).

The working principles in modelling a single neural mass model or a neural population are as follows: 1) a neural population has an average membrane potential, u, which is the result of different inputs received by a neural mass; 2) the received inputs are represented as average pulse densities (the average firing rate) which can be either inhibitory (from other neural masses in the model) or excitatory (from external sources outside the model); and 3) the output of a neural mass is also represented as an average pulse density. In the process of generating an output of a neural population, two main transformations - post-synaptic-potential (PSP) and potential-to-rate (e.g. using the Sigmoid function) - are adopted. Post-synaptic potential (PSP) is a temporary change in the electric polarization of a neuron. As a result of chemical transmission of a nerve impulse at the synapse, the PSP can lead to a firing of a new impulse (Purves et al. 2001) An average pulse density which represents the input to the model is converted to a membrane potential by the PSP transformation function. Each of the obtained potentials are multiplied by constant values representing the average number of synapses into the population.

The average membrane potential (*u)* of the model is achieved by summing all the excitatory inputs and subtracting the inhibitory ones. Finally, the average membrane potential is converted to an average pulse density using the potential-to-rate function.

PSP is a linear transformation and defined by a second-order differential equation which is characterised by its impulse function. Lopes da Silva (Lopes da Silva et al. 1976) first modelled the impulse response of a real PSP. Van Rotterdam (van Rotterdam et al. 1982) simplified it using Eq. 2.3 (Hebbink 2014). Here, α and β are constants and different in excitatory and inhibitory cases. α represents the maximal amplitude of PSP and is in millivolts, while, β is in s^{-1}.

$$h(t) = \begin{cases} \alpha\beta t e^{-\beta t} & t \geq 0 \\ 0 & t < 0 \end{cases} \qquad (2.3)$$

The second transformation is the sigmoid function which represents a nonlinear component in the model. Sigmoid is the gain function and in charge of converting the average membrane potential of a neural population (*u)* into an average pulse density. The sigmoid function is defined in Eq. 2.4 (Hebbink 2014) . e_0 is half of the maximal firing rate of the population. v_0 is the average membrane potential for which half of the population fires. *r* represents the steepness of the sigmoid function.

$$S(u) = \frac{2e_0}{1+e^{r(v_0-u)}} \qquad (2.4)$$

The common functions in most of the proposed neural mass modelling approaches are defined in Eq. 2.3 and Eq. 2.4 and are explained briefly in section 2.6.1.

The schematic diagram of these transformations is descripted in Figure 2.8 (Hebbink 2014).

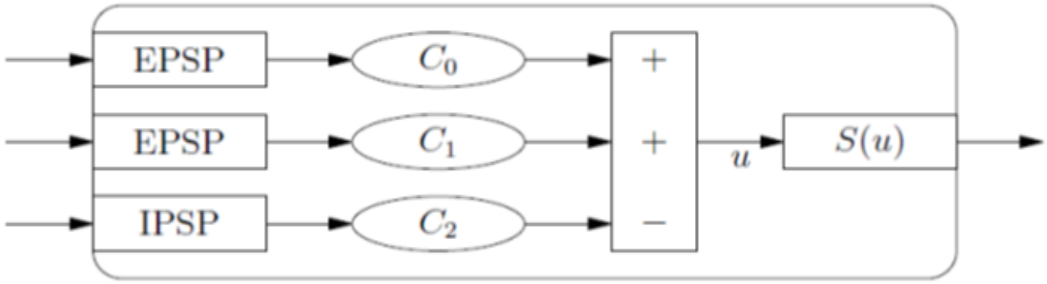

Figure 2.8 A schematic diagram of a neural mass (Hebbink 2014)Excitatory and inhibitory inputs to the model are represented as the average pulse density and are converted into the membrane potentials using PSP (i.e. either EPSP or IPSP respectively). The potentials are multiplied by constant values (C) which represent the average number of synapses coming to the neural population. The average membrane potential (u) is computed by summing up all the excitatory potentials and subtracting the inhibitory ones. Finally, the average membrane potential is converted into an average pulse density using the sigmoid function.

2.6.1 Neural Mass Modelling Approaches

2.6.1.1 Lopes da Silva's neural mass model

In this work (Lopes da Silva et al. 1974), the α-rhythm activities was investigated by modelling the thalamic module. The thalamic module consists of two neural masses, namely the thalamic cortical relay (TCR) cells and interneurons (INs); the TCR cell receives an external excitatory input and an inhibitory input from INs. INs only receive an excitatory input from the TCR cell. The transformation of the average membrane potential to an average pulse density was carried out using Eq. 2.3 and Eq. 2.4. A simplified schematic diagram of a thalamic neural mass model is illustrated in Figure 2.9 (Hebbink 2014).

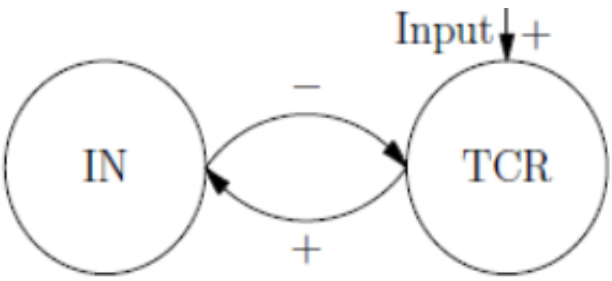

Figure 2.9 A schematic diagram of a neural mass model of the thalamus module (Hebbink 2014). The positive signed arrow represents the excitatory input and the negative signed arrow represents the inhibitory input.

2.6.1.2 Jansen and Rit's neural mass model of the cortical module

Jansen's neural mass model (Ben H. Jansen, Zouridakis, and Brandt 1993, B H Jansen and Rit 1995) is based on the work of Lopes Da Silva (Lopes da Silva et al. 1974, Lopes da Silva et al. 1976) and Van Roterdam (van Rotterdam et al. 1982). In this work (B H Jansen and Rit 1995), they developed the model further following a biologically inspired mathematical framework to simulate the recorded EEG signals, in particular the α-rhythm activities. The cortical module consists of three neural populations, namely pyramidal neuron (the main population), excitatory and inhibitory INs. The pyramidal cell is the only population that receives not only the external excitatory input, but also the inhibitory feedback from the inhibitory INs and the excitatory feedback from the excitatory INs. Both the inhibitory and excitatory INs receive the excitatory input from the pyramidal cell.

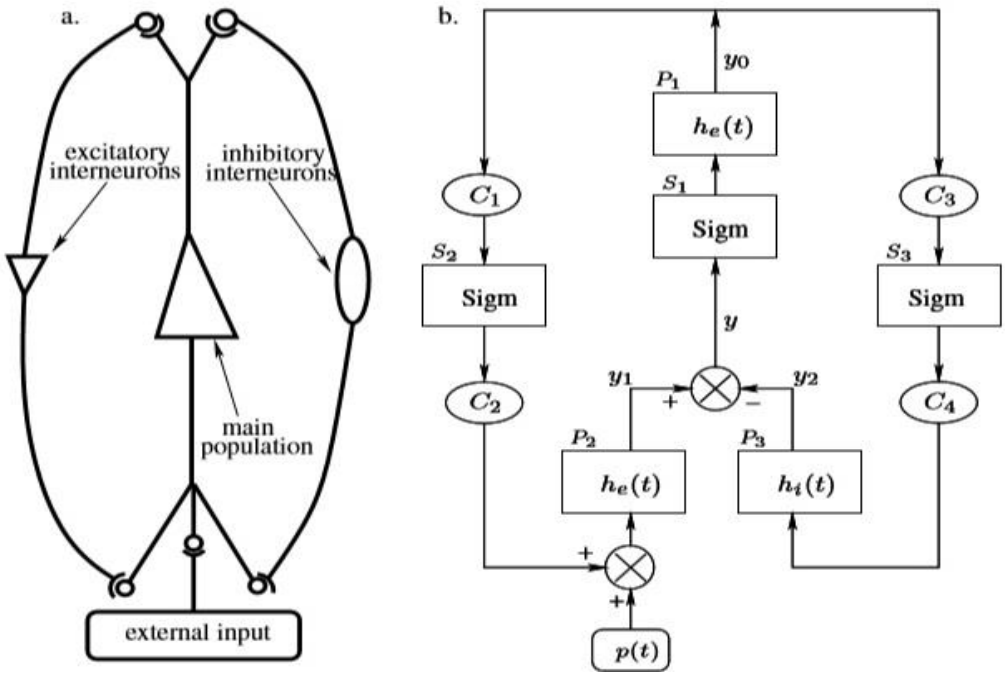

Figure 2.10 (a) A neural mass model of cortical module proposed by Jansen and Rit (Grimbert and Faugeras 2006). The model shows the interaction of the pyramidal cell with the excitatory and inhibitory INs. (b) Block presentation of cortical module with the mathematical operations performed in this module.

In Figure 2.10, the schematic diagram of the proposed neural mass model is illustrated (Grimbert and Faugeras 2006). In which, $p(t)$ represents the external excitatory input to the model; the post-synaptic-potentials p_i, where $i = 1,2,3$, are modelled with $h_i(t)$ or $h_e(t)$ which convert the average pulse density to an average excitatory or inhibitory post-synaptic-potentials using Eq. 2.3; y_0, y_1 and y_2, represent the outputs of three post-synaptic blocks, and the sigma blocks represent the sigma function defined in Eq. 2.4 which converts the average membrane potential to an average pulse density as the model's outputs; the constants C_i represent the strength of the synaptic connections between the populations.

According to the PSP function defined in Eq. 2.3, if $x(t)$ is the input to the system, $y(t)$ is the convolution product $h * x(t)$. Therefore, the corresponding differential equation can be defined as in Eq. 2.5 (Ben H. Jansen, Zouridakis, and Brandt 1993, Grimbert and Faugeras 2006).

$$\ddot{y}(t) = \alpha\beta x(t) - 2\beta\dot{y}(t) - \beta^2 y(t) \qquad (2.5)$$

In the excitatory case, α and β in Eq. 2.5 can be specified as A and a respectively. Where in the inhibitory case, α and β are represented by B and b respectively. The second-order differential equation can be redefined as a system of two first-order equations as defined in Eq. 2.6 (B H Jansen and Rit 1995, Grimbert and Faugeras 2006),

$$\begin{cases} \dot{y}(t) = z(t) \\ \dot{z}(t) = \alpha\beta x(t) - 2\alpha z(t) - \alpha^2 y(t) \end{cases} \qquad (2.6)$$

As it is mentioned earlier, y_0, y_1 and y_2 represent the outputs of three post-synaptic blocks and their derivatives are noted as y_3, y_4 and y_5; writing two equations similar to Eq. 2.6 for each post-synaptic-potentials results in six first-order differential equations. The dynamics of Jansen and Rits's model are explained by the differential equations as defined in Eq. 2.7 (B H Jansen and Rit 1995, Grimbert and Faugeras 2006).

$$\begin{cases} \dot{y}_0(t) = y_3(t) \\ \dot{y}_1(t) = y_4(t) \\ \dot{y}_2(t) = y_5(t) \end{cases} \qquad (2.7)$$

$$\begin{cases} \dot{y}_3(t) = AaSigm[y_1(t) - y_2(t)] - 2ay_3(t) - a^2 y_0(t) \\ \dot{y}_4(t) = Aa\{p(t) + C_2 Sigm[C_1 y_0(t)]\} - 2ay_4(t) - a^2 y_1(t) \\ \dot{y}_5(t) = BbC_4 Sigm[C_3 y_0(t)] - 2by_5(t) - b^2 y_2(t) \end{cases}$$

The model's output is represented by the potential of the pyramidal population ($y = y_1 - y_2$) as it is believed that the pyramidal cell in the cortex has the biggest influence on the EEG signal (Kandel 2000).

2.6.1.3 Wendling's neural mass model of the cortical module

In this work (Wendling et al. 2002), the focus has been placed on studying high-frequency EEG activities (i.e. the gamma band) which is the most characteristic electrophysiological pattern seen in focal seizures of people affected by epilepsy. According to (White et al. 2000, Whittington et al. 2000), the generation of gamma activities are linked to the behaviour of inhibitory INs, particularly, two types of Gamma-Aminobutyric-Acid ($GABA_{A,slow}$,and $GABA_{A,fast}$) inhibitory post-synaptic-potential (IPSP). Therefore, a new computational macroscopic model of EEG activities is introduced to include a physiological relevant fast inhibitory feedback loop. The cortical module is modelled with four subsets of populations including: pyramidal cells, excitatory INs, inhibitory INs with slow synaptic kinetics known as $GABA_{A,slow}$, and inhibitory INs with fast synaptic kinetics known as $GABA_{A,fast}$. Regarding these two types of inhibitory INs, $GABA_{A,fast}$ is near the soma and is a rapidly activated and decaying IPSP mediated by somatic synapses. $GABA_{A,slow}$ is on the dendrites and is a slow rising and decaying IPSP mediated by dendrites synapses. Therefore, the two classes of inhibitory INs give rise to the two mentioned IPSPs and mainly contribute to the generate gamma rhythms.

Figure 2.11 (Wendling et al. 2002) shows the modelling of the cortical module, in which the pyramidal cell receives an external excitatory input $p(t)$, representing the influence from neighbouring areas. $p(t)$ is modelled by Gaussian white noise. The pyramidal cell also receives an excitatory input from the excitatory INs and inhibitory input from the two classes of inhibitory INs. The excitatory INs and both classes of inhibitory INs receive excitatory feedbacks from the pyramidal cell. The output of the model is the summated post-synaptic potentials in the pyramidal cell and is translated as an EEG signal.

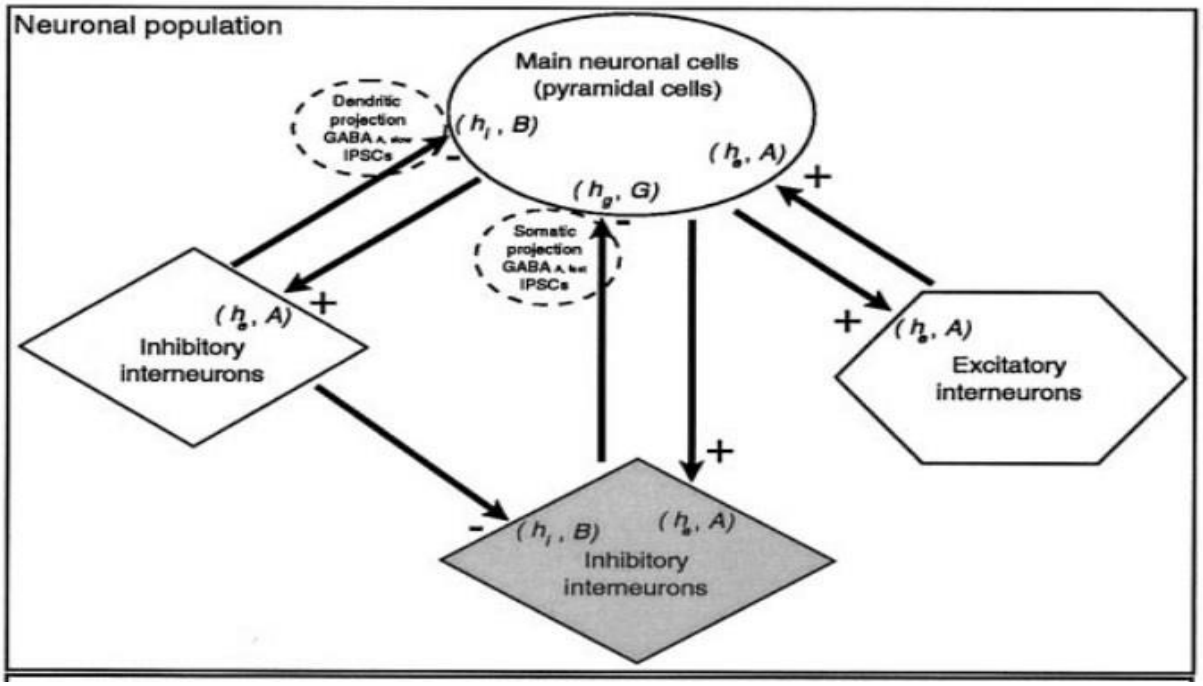

Figure 2.11 Neural population modelling of the cortical module by Wendling (Wendling et al. 2002).

In Figure 2.12, the mathematical operations performed in the cortical module is illustrated with four subsets, the pyramidal cell, excitatory INs and two classes of inhibitory INs (Wendling et al. 2002).

The 2nd order linear transfer function converts the average pulse density of the input to an average post-synaptic membrane potential for either slow inhibitory, excitatory or fast inhibitory with respective impulse response, $h_e(t) = A\,a.e^{-at}$, $h_i(t) = B\,b.e^{-bt}$, and $h_g\,(t) = G\,g.e^{-gt}$, where, A, B and G represent the synaptic gains. , and represent time constants in excitatory, slow inhibitory and fast inhibitory cases.

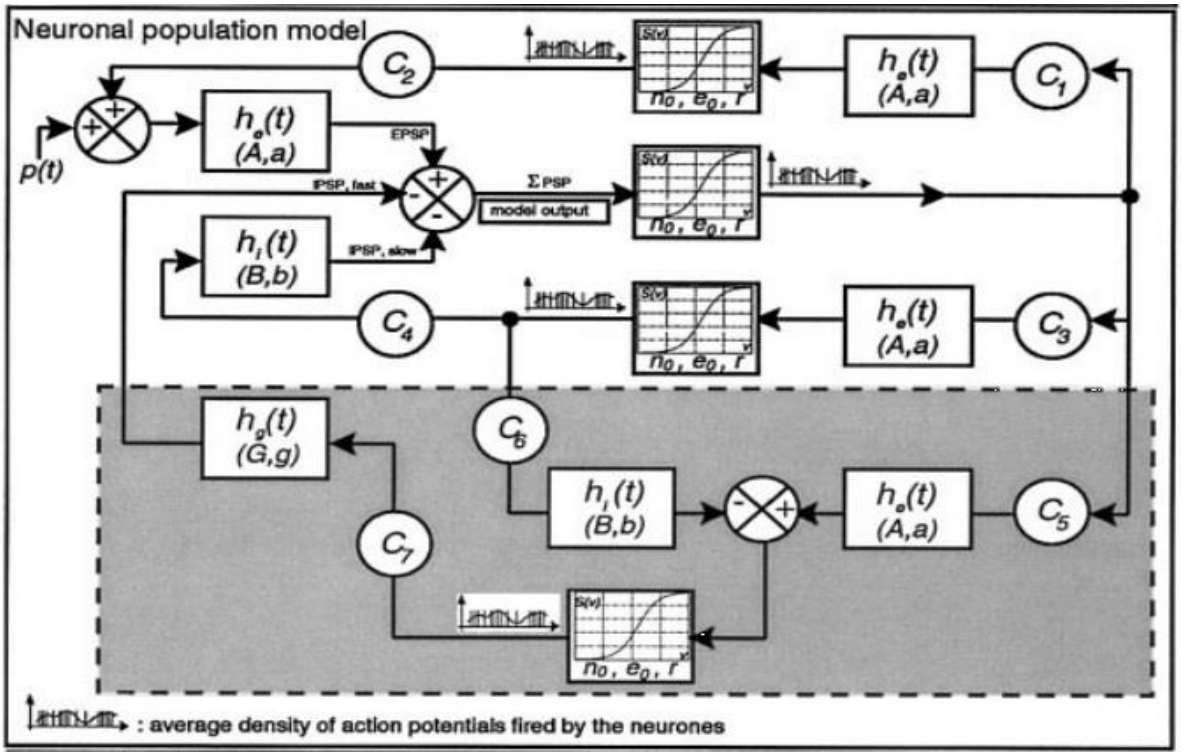

Figure 2.12 The computational model of the cortical module presented in Figure 2.7 (Wendling et al. 2002).

For transferring the average postsynaptic potential to an average pulse density potential, a static nonlinear *s (v)* function defined in Eq. 2.4 is used. C_1-C_7 , represent the average number of synaptic connectivity between the four populations. As mentioned in Jansen and Rit's modelling, each linear transfer function introduces a pair of the first order differential equations as defined in Eq. 2.8 (Wendling et al. 2002), where $W = A$, $W = B$, $W = G$, and $w = a$, $w = b$, and $w = g$ represent the excitatory, slow inhibitory and fast inhibitory cases respectively; $x\ (t)$ and $z(t)$ represent the input and output of the linear transfer function respectively.

$$\dot{Z}_1(t) = z_2(t)$$
$$\dot{z}_{2(t)} = wgx(t) - 2wz_2(t) - w^2 z_1(t) \qquad (2.8)$$

Writing differential equations defined in Eq. 2.8 for each subset results in a set of ten differential equations showed in Eq. 2.9 (Wendling et al. 2002) that govern the dynamics of the

Wendling's population model. This set of equation is solved by classical numerical integration methods (i.e. Runge-Kutta) (Honeycutt 1992).

In summary, Wendling's model is an extended version of Jansen and Rit's work in which a new class of inhibitory INs with a fast-synaptic kinetics is introduced.

$$\dot{y}_0(t) = y_5(t)$$

$$\dot{y}_5(t) = AaS[Y_1(t) - y_2(t) - y_3(t)] - 2ay_5(t) - a^2y_0(t)$$

$$\dot{y}_1(t) = y_6(t)$$

$$\dot{y}_6(t) = Aa\,\{p(t) - C_2S[C_1y_0(t)]\} - 2ay_6(t) - a^2y_1(t) \tag{2.9}$$

$$\dot{y}_2(t) = y_7(t)$$

$$\dot{y}_7(t) = BbC_4S[C_3y_0(t)] - 2by_7(t) - b^2y_2(t)$$

$$\dot{y}_3(t) = y_8(t)$$

$$\dot{y}_8(t) = GgC_7S\left[C_5y_{0(t)} - C_6y_4(t)\right] - 2gy_8(t) - g^2y_3(t)$$

$$\dot{y}_4(t) = y_9(t)$$

$$\dot{y}_9(t) = BbS[C_3y_0(t)] - 2by_9(t) - b^2y_4(t).$$

It is believed the both classes of inhibitory INs significantly contribute to the generation of high-frequency EEG signals (the gamma frequency band). The model showed that it is capable of generating six different types of EEG activities in close resemblance to the real EEG activities recorded interictally or ictally (in EEG, the recording during a seizure is referred as ictal, and interictal represents the period between seizures). These six types include: Types 1 and 2 representing the normal background activities and sporadic spikes which are observed in real signals during interictal periods, Types 3 and 4 representing the sustained spikes and slow

rhythmic activities, both seen during seizures, Types 5 representing low-amplitude discharges usually appeared at the start of ictal periods, and finally Type 6 representing slow quasi-sinusoidal which resembles the real ictal activity which often follows the rapid ictal activity in time. Producing these types of activities depends on variations of the model parameters such as A, B, G, which respectively represent the excitatory, slow inhibitory and fast inhibitory synaptic gains from INs to the pyramidal cell. The visual inspection of the results and also estimating the power spectral feature show that the model is capable of simulating realistic epileptiform pattern even without carrying out the precise curve fitting in the simulation.

2.6.1.4 Basadatta's neural mass model of the thalamo-cortico-thalamic circuit.

This work (Basabdatta Sen Bhattacharya, Coyle, and Maguire 2011b) presents a computational model of thalamo-cortico-thalamic (TCT) circuit which consists of a thalamic module and a cortical module in order to simulate the alpha rhythms and study EEG abnormality associated with the Alzheimer's disease (AD). It is known that TCT plays a vital role in the generation of brain rhythms (Basabdatta Sen Bhattacharya, Coyle, and Maguire 2011b). The TCT loop is structured as two parts, namely the thalamus and cerebral cortex, in which neural pathways connect them together in a loop format. Thalamic nuclei are populated by two types of neurons, namely the Thalamo-cortical relay cells (TCR) and the IN cells. Thalamic Reticular Nucleus (TRN) is located in the way that all the thalamo-cortical and cortico-thalamic axons pass through it and make excitatory synapses on its cells.

In the cortical module, there are four types of cells, the pyramidal cells (PY), Excitatory INs (eINs), slow inhibitory INs (sINs) and fast inhibitory INs (fINs). TCR and IN populations receives excitatory inputs from the retinal and cortical cells; the IN and TRN cells make inhibitory synapses on the TCR cells. The interaction of the pyramidal cell in the cortical module

with three other subsets is the same as it is explained in Wendling's model (Wendling et al. 2002) except having additional excitatory synapses connected to the pyramidal cell from an external source and the TCR cell in the thalamic module. The Brain Reticular Formation (BRF) is another major source of the input to thalamic and cortical modules, which makes excitatory synapses on the TCR cells and inhibitory synapses on the IN and TRN cell populations. Figure 2.13 (Basabdatta Sen Bhattacharya, Coyle, and Maguire 2011b) illustrates the synaptic organisational structure in the TCT circuit.

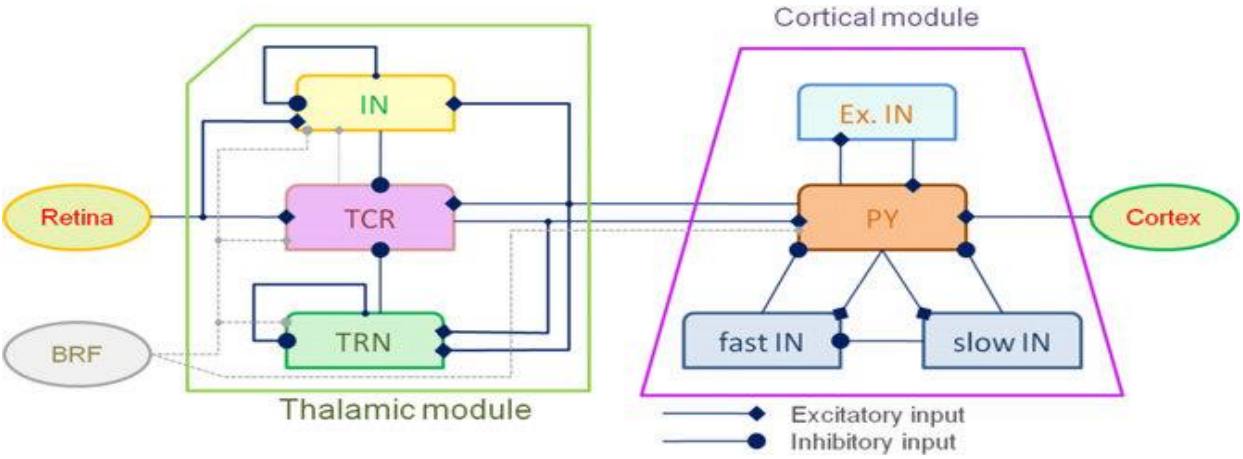

Figure 2.13 The synaptic organisational structure in the thalamo-cortico-thalamic circuit (Basabdatta Sen Bhattacharya, Coyle, and Maguire 2011b).

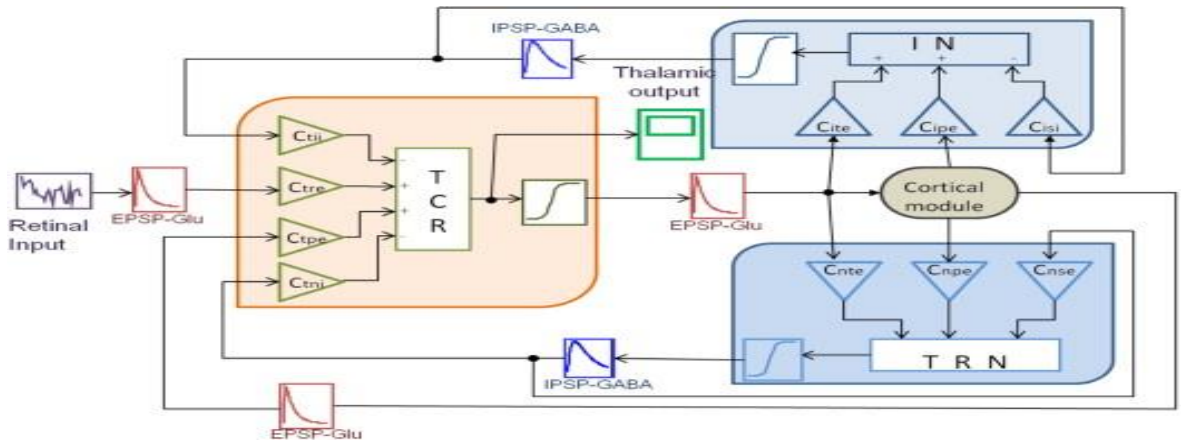

Figure 2.14 The computational model of the thalamic module (Basabdatta Sen Bhattacharya, Coyle, and Maguire 2011b).

In Figure 2.14, the computational model of the thalamic module is shown. The excitatory post-synaptic-potential (EPSP) and inhibitory-post-synaptic-potential (IPSP) describe synaptic response as defined in Eq. 2.10 (Basabdatta Sen Bhattacharya, Coyle, and Maguire 2011b), where, $\overline{w}$ represents the inhibitory or excitatory synapse; $\tau_{\overline{w}}$ indicates the delay in the rise time of the synaptic function and the amplitude of the synapse is presented by $H_{\overline{w}}\,(t)$. The membrane potential is converted to an average pulse density using the sigmoid function as defined in Eq. 2.4. C parameters are constant values representing the average number of synapses between the populations.

$$h_{\overline{w}}\,(t) = \frac{H_{\overline{w}}}{\tau_{\overline{w}}}\; te^{-\frac{t}{\tau_{\overline{w}}}} \qquad (2.10)$$

The extrinsic input to the cortical module is from the neighbouring cortical voxels simulated with white Gaussian noise. The computational model of the cortical module disconnected from the thalamic module is shown in Figure 2.15.

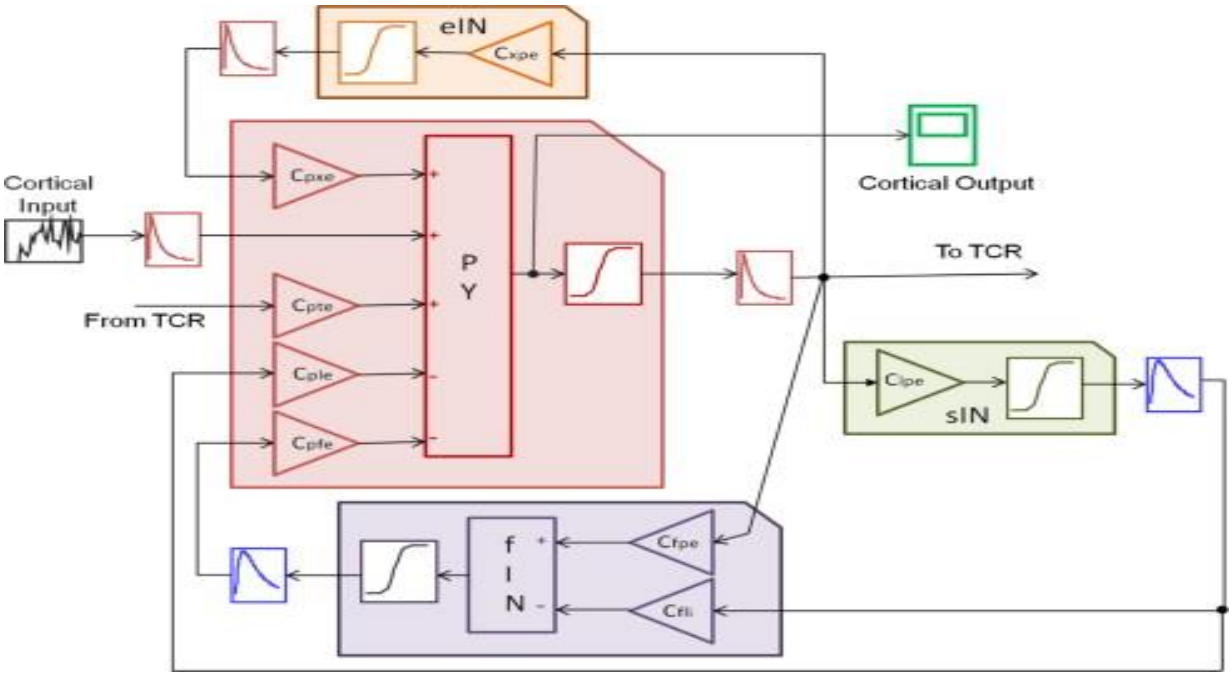

Figure 2.15 The computational model of the cortical module (Basabdatta Sen Bhattacharya, Coyle, and Maguire 2011b).

The results showed that the model is capable of generating the alpha rhythms. By varying the synaptic connectivity parameters (C) in the thalamic module, the model can simulate the effects of AD on the brain synaptic circuitry as the power spectrum within the alpha rhythms can be observed.

2.6.1.5 Induced activities modelling

Regarding the modelling of induced activities, computational model of thalamo-cortical networks was used (Pfurtscheller and Lopes da Silva 1999, Neuper and Pfurtscheller 2001, Basab Bhattacharya, Coyle, and Maguire 2012) to investigate the mechanisms underlying the event known as focal ERD surrounded ERS within the alpha band. It is observed (Pfurtscheller and Lopes da Silva 1999) that in an experiment of paying attention to a specific sensory modality the corresponding alpha activity decreases, which is known as desynchronisation of the alpha activity. For example, when a subject is instructed to make a particular task such as cue-guided movement of the hand or the foot, the power spectral within the alpha band at the focal area of the brain will be decreased. While, in the neighbouring area of the brain that corresponds to the same or other modalities of information processing, an increase in the power spectral within the alpha band will be observed. This event is known as focal ERD surrounded ERS within the alpha range.

An example of modelling a thalamo-cortical network for ERD/ERS events is presented in (Basab Bhattacharya, Coyle, and Maguire 2012). In this work, the aim is to investigate the ERD/ERS behaviours of the subjects affected by AD within the alpha band. It is believed that in AD patients, there is an increased latency in the cortical response corresponding to an event which affects their ERD/ERS behaviours compared to young healthy adults (Basab Bhattacharya, Coyle, and Maguire 2012). The reason for modelling the thalamo-crotical network for this purpose can be justified by considering several facts: 1) it is believed that

ERD/ERS events are related to the task that require attention such as visual perception or limb movement and it is the thalamus which links perception to the action; 2) all sensorimotor pathways have branches that are afferents (in the nervous system, afferent nerves carry nerve impulses from the sense organs toward the nervous system) to the thalamic nuclei; and 3) studies show that thalamus plays a vital role in generating the cortical alpha band (Murray Sherman and Guillery 2001). Figure 2.16 (Basab Bhattacharya, Coyle, and Maguire 2012) illustrates a single model of the thalamo-cortical network, including two subsets: the thalamo-cortical-relay (TCR) and the thalamic-reticular-nucleus (TRN). The TCR cell receives an excitatory input from the retinal which is simulated by the white Gaussian noise.

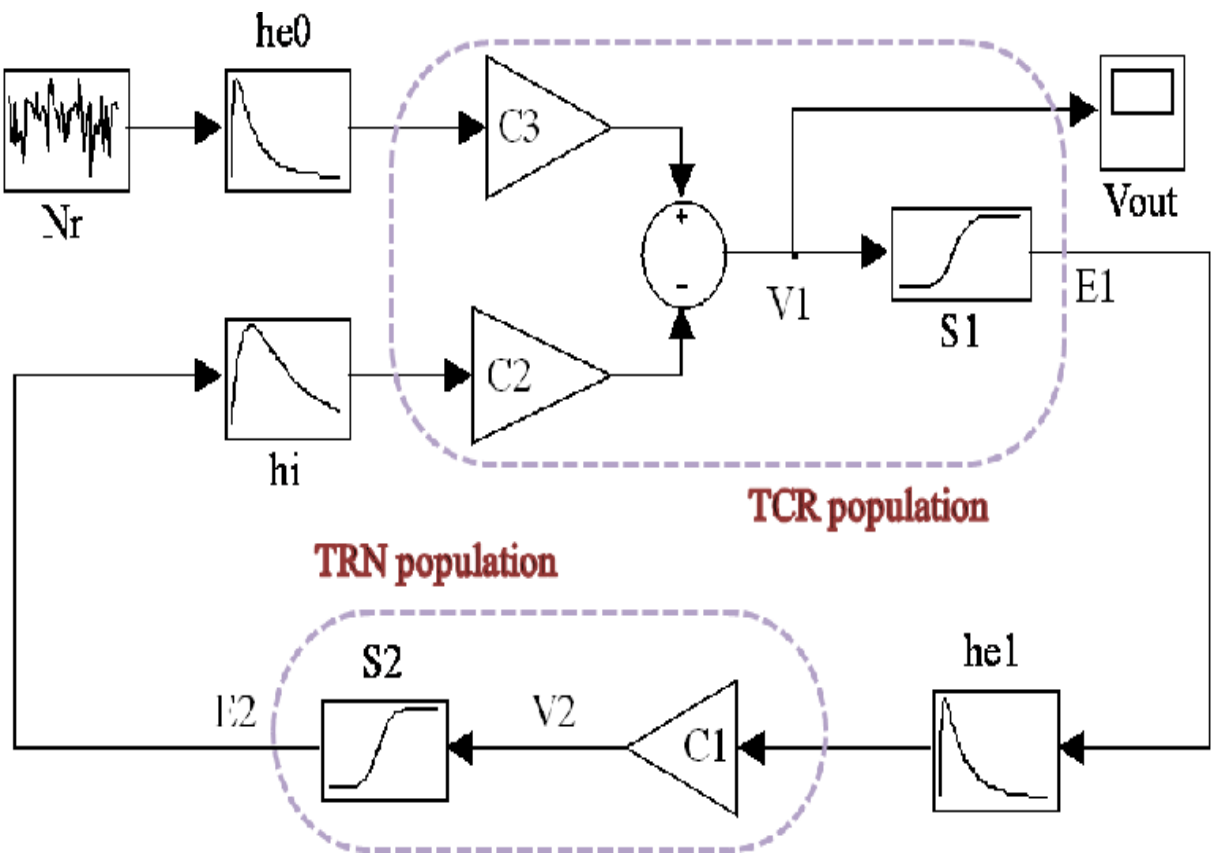

Figure 2.16 The computational model of the thalamo-cortical network including TCR and TRN populations (Basab Bhattacharya, Coyle, and Maguire 2012).

The synaptic response is described by $h(t)$ as defined in Eq. 2.3. e and i are excitatory and inhibitory synapses respectively. The average membrane potential is converted to an average pulse density $E(t)$, using the sigmoid function as defined in Eq. 2.4. C represents the average number of synaptic connectivity between the populations.

In order to simulate ERD/ERS events, multiple cascaded models were used which is shown in Figure 2.17 (Basab Bhattacharya, Coyle, and Maguire 2012).

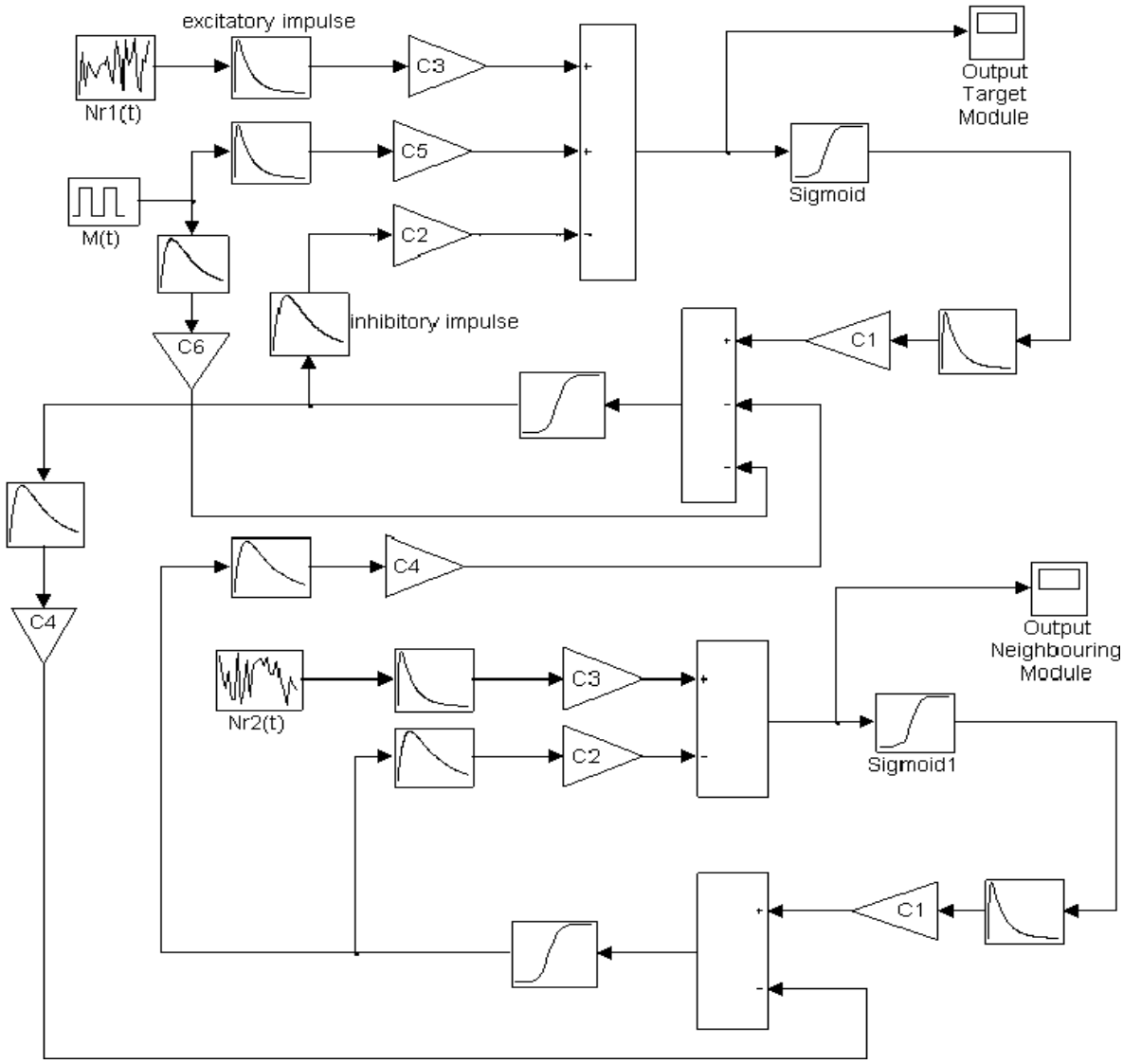

Figure 2.17 The computational model of two thalamo-cortical networks in order to simulate the generation of ERD/ERS events. The model consists of a target module (the top model) and a neighbouring module (Basab Bhattacharya, Coyle, and Maguire 2012).

The above computational model consists of two thalamo-cortical networks: 1) a target module, and 2) a neighbour module. The two modules are connected through inhibitory connectivity parameter C_4, simulating the mutual inhibition in the neighbouring population of thalamo-cortical neurons. $M\ (t)$ is a modulatory input which simulates the cholinergic input to the thalamus from the brainstem companied with an event such as visual, mental, or motor activity tasks. The modulatory input has an excitatory connectivity with the TCR cell population through C_5 and inhibitory connectivity with the TRN cell population through C_6 in the target module. It is worth mentioning that the neighbouring module does not receive direct afferents from the modulatory input. In the target module, the modulatory input caused a reduction in the power and synchronisation. While in the neighbouring module, a power enhancement and increasing in synchronisation were observed.

In summary, the above model is capable of generating focal ERD surrounded ERS events.

2.6.2 Parameterisation (Data-fitting) Techniques for Neural Mass Models

There are a large number of parameters in the differential equations of the NMMs discussed in Section 2.6.1. Therefore, parameterisation methods are used to find the best combination of parameter values so that the NMMs can reproduce the recorded EEG data. In the following, relevant parameterisation techniques are explained.

2.6.2.1 Kalman Filters-based Approaches

One of the methods for parameterisation of NMMs is based on Kalman Filters (KF). Extensions and generalisations to KF have also been developed, such as Extended Kalman Filter (EKF), Unscented Kalman Filter (UKF) and Cubature Kalman Filter (CKF). KF algorithm (Kalman

1960) is a recursive filter that can be applied to a linear dynamic system in order to obtain the estimation of the model's parameters. It is a two-step procedure containing prediction and update (correction). In the prediction step, at each discrete time increment, a linear operator is applied to the previous state to generate the new state with some noise mixed in. Then, a further linear operator mixed in with more noise generates the observed output from the hidden or true state. In the update step, the estimation is updated based on the computed Kalman gain which is the weighted mean representing the certainty about the estimation. The recursive characteristic of this algorithm makes it possible to run in real time as it only needs the best previous state estimation, present input measurements and the uncertainty matrix, rather than the entire history.

Due to the fact that KF cannot handle nonlinearities, EKF is introduced (Einicke and White 1999) which also follows the predictor/corrector steps of KF. However, the state and observation models in EKF are nonlinear functions. EKF (Einicke and White 1999) performs linearization around the best answer from the previous point by computing the first derivative of the state and observation functions (Jacobian matrix) at each time step. Although claimed to be able to handle nonlinearities, the algorithm suffers from the following constraints: a) the system functions should be differentiable; b) only moderate nonlinearities and uncertainty can be handled; c) it cannot cope with high dimensional problems; d) it is sub-optimal and biased estimator due to the linearization step; and e) computing the Jacobian matrix is an error-prone and computational expensive process.

Based on the fact that EKF cannot handle highly nonlinear systems as well as non-differentiable functions, Julier and Uhlmann (Julier and Uhlmann 1997) proposed UKF with a better accuracy and easier implementation compared to EKF. UKF does not need to linearize the nonlinear functions and compute the Jacobian matrix. In UKF, a set of sample points called sigma points are carefully chosen around the mean via unscented transform technique. These

sigma points are then propagated through the nonlinear functions so that the posterior estimation can be obtained accurately. Although UKF can handle nonlinear and non-differentiable systems, it requires the prior knowledge about the noise-condition (Mean and variance) that is added to the measurement model which is still a constraint likewise for other types of KF methods (KF, EKF, UKF).

CKF is another type of KF methods proposed by Arasaratnam and Haykin which is based on spherical-radial cubature rule (Arasaratnam and Haykin 2009). The cubature rules help to address the multi-dimensional integral issues. In general, CKF is preferred over UKF due to its more stable performance. However, since CKF used the third-degree spherical-radial cubature rule, it is not as accurate as UKF in real-life applications. Furthermore, computing the formula can cause inconveniences in high-dimensional state-estimation problems (Xin-Chun Zhang 2013).

More details of using the KF-based approaches for parameterisation of neural mass models can be found in Chapter 6.

2.6.2.2 Least Square based Approaches

Least squares based approaches follow a standard approach in regression analysis to approximate the solution of a system by minimizing the sum of the squared differences between the fitted and observed values. As the number of observations (samples) is normally bigger than the number of unknowns, the system to be approximated is overdetermined. The minimum of the sum of squares is found by setting the gradient to zero. Therefore, if the model contains *m* independent variables, there are *m* gradient equations to work out the best value for the coefficient pertaining to each independent variable.

The most important application is in data fitting. A simple example is the linear regression in which the coefficients of the regression equation are 'trained' through the available data. For complex systems, the linear regression may not be sufficient, which leads to the need for the non-linear regression techniques .Furthermore, simple regression and least-squares methods are inefficient if the problem has substantial uncertainties in the independent variable. In such cases, instead of fitting errors in the least squares sense, a modified fitting procedure to minimise the errors in variables is needed.

More details of using the least-squares based approaches for NMM parameterisation can be found in Chapter 6.

Melissa Zagvalia's work. In this work (Zavaglia et al. 2006), a neural mass model of the cortical module based on Wendling's work is used. In terms of modelling a Region of Interest (ROI), three populations named as POPLF, POPMF and POPHF are connected in parallel to generate rhythms in low, mid and high frequency bands respectively. Each population receives input simulated as the white Gaussian noise with different variances. Before the fitting procedure, real data were recorded from two subjects performing finger movements of the right hand and the execution of a working memory task. In Figure 2.18, the parallel arrangement of the three populations in a ROI is shown.

$\boldsymbol{p}_1$ represents the input (white the Gaussian noise) to each population. $\boldsymbol{v_{out}}$ is the cortical EEG of the ROI and is obtained as the mean value of the membrane potentials $\boldsymbol{v}_0$ of pyramidal neurons in three populations, which is defined in Eq. 2.11 (Zavaglia et al. 2006).

$$v_{out}(t) = 1/3 \sum_{K=L,M,H} v_0^k\,(t) \qquad (2.11)$$

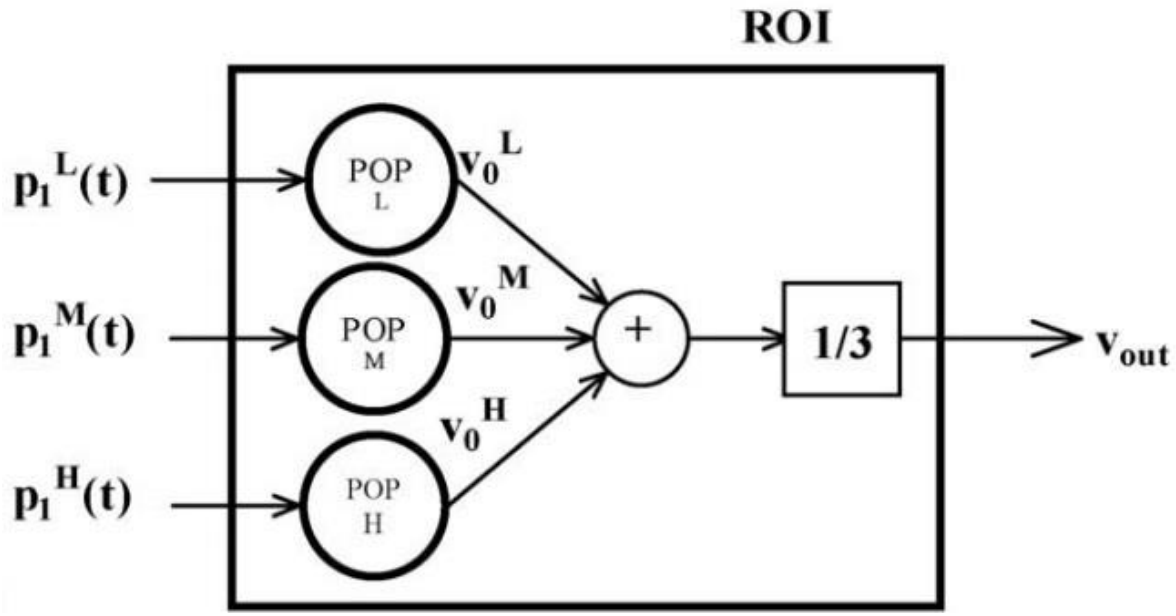

Figure 2.18 Parallel arrangement of three populations in a ROI of the cortical module (Zavaglia et al. 2006).

In the fitting procedure, the simulated EEG of the ROI and real data were compared in frequency domain. The minimisation of the least squared difference between the power spectrum densities (PSDs) in the range of 3-50 HZ was carried out. Parameters involved in parameterisation procedure include the mean values, standard deviation and values of fast inhibitory average gains of synapses for all three populations.

It is shown that the model can simulate different EEG rhythms by only modifying the time constants parameters and fitting the positions of individual rhythms by modifying the average gain of fast inhibitory synapses. To improve the model in order to capture the complexity of the cortical EEG, three populations were connected in parallel. This fitting procedure suffers the following problems: a) a large number of parameters and increased complexity of the model in order to simulate the dynamics of a single ROI; therefore, the accuracy of the parameterisation will be affected as least squares method is not performing well in those complex situations; b) no physiological explanation supported the obtained results. It is only

demonstrated that the model can achieve a precise match for the power spectra; and, c) the parameterisation procedure is based on the linear assumption while in reality neural systems are characterised by their nonlinearity.

A simplified version of the model (Zavaglia et al. 2008) involving a small number of parameters was developed. However, the other two of the aforementioned issues still remained. Furthermore, the parameterisation procedure was carried out in a trial-and-error manner.

2.6.2.3 Genetic Algorithm (GA) based Approaches

Genetic Algorithms (GAs) were originally proposed by John Holland (Holland 1992) in order to find a good solution to a given optimisation problem. They are general purpose search algorithms which use principals inspired by natural selection and survival of the fittest to evolve solutions to problems. GAs start the process on the randomly generated population of individuals and evolve that population through operations such as selection of the individuals as the parents for the next generation, combination of parents to form the children of the next generation, and random changes to the population of the next generation (Goldberg 1989). The next generation is generated by applying these operations. For each generation, a collection of chromosomes which are representations of individuals within the population are evaluated against the objective functions. As this process iterates, the population evolves toward the optimal solution. In many areas and specifically in the field of artificial intelligence, GAs are considered as one of the most viable approaches to find the near global optima of a complex problem. The algorithm does not require any prior knowledge of the problem.

GAs are particularly suitable in dealings with the Multi-objective optimisation problems, where there are conflicting objectives and the interaction between different objectives leads to a set of compromised solutions which are known as Pareto-optimal solutions (Savic 2002). The

existence of a pool of candidate solutions and the parallel processing mechanism make GAs an ideal candidate to find Pareto-optimal solutions at the same time, without converting the MOP problem into the SOP problem. Another two notable advantages of GAs lie in their ability in dealing with ill-defined objective functions without the need of having close-form and differentiable objective functions and provide near optimal solutions.

More details of using the GA-based approaches for parameterising NMM and generate fuzzy rule-based classifiers can be found in Chapters 4, 5 and 6.

2.7 Conclusion

In this chapter, basic of neuroscience and the main principals of BCI research, machine learning algorithms (in particular linear and non-linear classification algorithms), neural mass modelling approaches, and different techniques including least squares and metaheuristic based approaches for parameterising NMMs and eliciting nonlinear classifiers are reviewed. The techniques described in this chapter are by no means exhaustive. In the following chapters, the implementations of the mentioned techniques are presented. In particular, Chapters 4 and 5 describe a multi-objective immune algorithm for eliciting fuzzy rule-based and ensemble classifiers, and Chapter 6 presents a GA based approach for parametrising NMM

Chapter 3

Experimental EEG Data

The primary objective of this chapter is to introduce the process of the Electroencephalogram (EEG) data recording which have been used for creating the datasets for further experiments that are implemented in this project and are explained in the next chapters. In this chapter, the process of data acquisition has been explained followed by the pre-processing procedure for the collected EEG data. Finally, the pre-processed data were used in the feature extraction procedure to create the feature vectors for the classification algorithm developed in the next chapter.

3.1 EEG Data Acquisition

In this project, a non-invasive technique known as EEG has been used for the data acquisition procedure. Steady State Visual Evoked Potential (SSVEP) signal has been selected as the EEG-signal type among the other EEG-paradigms as have been explained in Chapter 2. SSVEPs demand not only less training, but also offer high information transfer rate (ITR) and are less sensitive to artefacts (X. Chen et al. 2015). SSVEPs are stable oscillations in voltage that are elicited by visual stimuli such as a light flash (J. Wolpaw and Wolpaw 2012). To acquire SSVEPs, a subject is provided with a visual field which represents constant visual stimuli. Each stimulus has a specific frequency and the subject is asked to fixate on a specific frequency. Due to the subject fixation on a specific stimulus, a peak will be seen in the frequency spectrum around the same frequency as that of the stimulus, as well as at its harmonics. SSVEPs are

normally present in a frequency range of 1Hz to 100 Hz (Lin et al. 2007). In the remaining part of this Section, the full procedure of EEG recording used in this study has been explained.

3.1.1 Subjects

Ten subjects, including seven males and three females within the age of 19-34 years-old, participated in the experiments. All the subjects had normal or corrected-to-normal visons. Subjects have been trained prior to the experiments. All experiments were conducted in accordance with the Declaration of Helsinki and the British Psychological Society guidelines. Ethical approval for the study was obtained from the School of Psychology Ethics committee, University of Lincoln.

3.1.2 Experiment Set-up

An MSI (MS-7788) computer, with i7-3990CPU Intel processor, NVida GeForce GTX 650 graphics card, and a 64-bit Windows 7 operating system, was used to create and display the stimuli. The stimuli were displayed using a 22-inch Illyama HM204DTA Vision Master Pro 514 Diamondtron U3-CRT monitor, with a Bits # signal processor (Cambridge Research Systems, Cambridge, UK), which was calibrated using an LS100 Minolta photometer. The maximum luminance of the display was $2.58cd/m^2$, and the minimum luminance of the display was $151.10cd/m^2$. The screen refresh rate was $85Hz$, and the resolution of the display was 1024×768 pixels. MATLAB 2015a (The Mathworks Inc., Natick) and the Psychtoolbox v3 functions (Brainard 1997, Pelli 1997, Kleiner et al. 2007) were used to generate and display the stimuli. A Biosemi 64-channel Active-Two system was used to record EEG signals. Electrodes were placed according to the 10-20 system (Malmivuo and Plonsey 1995) using an EEG cap. Eight additional electrode were placed on the right and left mastoids, infra and supra-

orbital, and on the outer canthi of the eyes. Electrically conducting gel was used to reduce impedance.

3.1.3 Data Recording

Subjects were seated 1m away from the display screen in an electrically insulated and sound-attenuated room. Subjects were asked to wear custom-made glasses to partially obscure their vision. Custom-made glasses used in this experiment with an effect of blurring vision to reduce effective contrast and to remove sharp luminance gradients was employed to approximate a 1-dimensional stimulus. The necessity of generating 1-dimensional stimulus in this research is due to the input-type requirement of the neural-mass-model that has been used for this project to reproduce the EEG signals. Figure 3.1 shows the EEG data recording from one of the subjects while he was sitting in an electrically insulated and sound-attenuated room during the experiment. Figure 3.2 shows the SSVEP signals generated as the responses to different stimuli.

At the beginning of the experiments, subjects were asked to close their eyes for two minutes and the alpha-brain activity signals were recorded. Subsequently, the main experiment of recording SSVEPs started. Subjects initiated each trial by pressing the SPACE bar. The stimuli were created by sinusoidally varying the contrast of the display, at frequencies of either 10Hz, 14Hz, or 21Hz, emulating flashing light. The presence of each individual stimulus (one trial) lasted 10 seconds. There were 5 repetitions for each stimulation frequency within a block. The stimuli within each block were presented in a random order to minimise the subject's anticipation of the next stimuli of a certain frequency. Each subject viewed a total of 13 blocks, with short breaks in between each block. Subjects were asked to remain as still as possible during the recording to minimise movement artefacts.

Figure 3.1 Subject is sitting in an electrically insulated and sound-attenuated room for the SSVEP recording experiment.

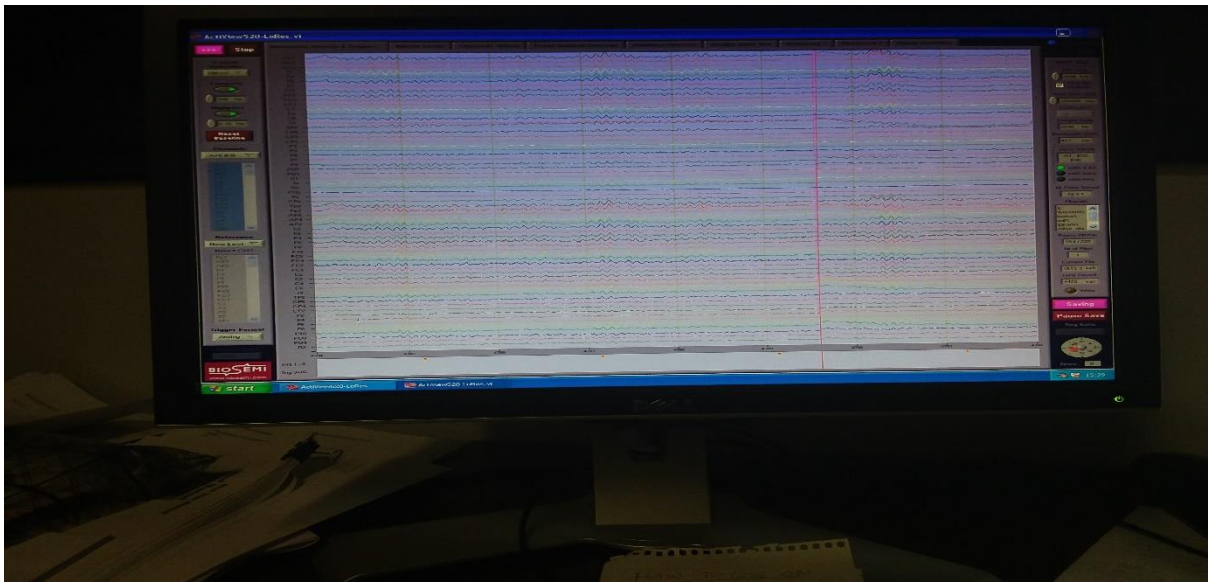

Figure 3.2 SSVEP signals are generated based on the responses to three different stimuli frequencies.

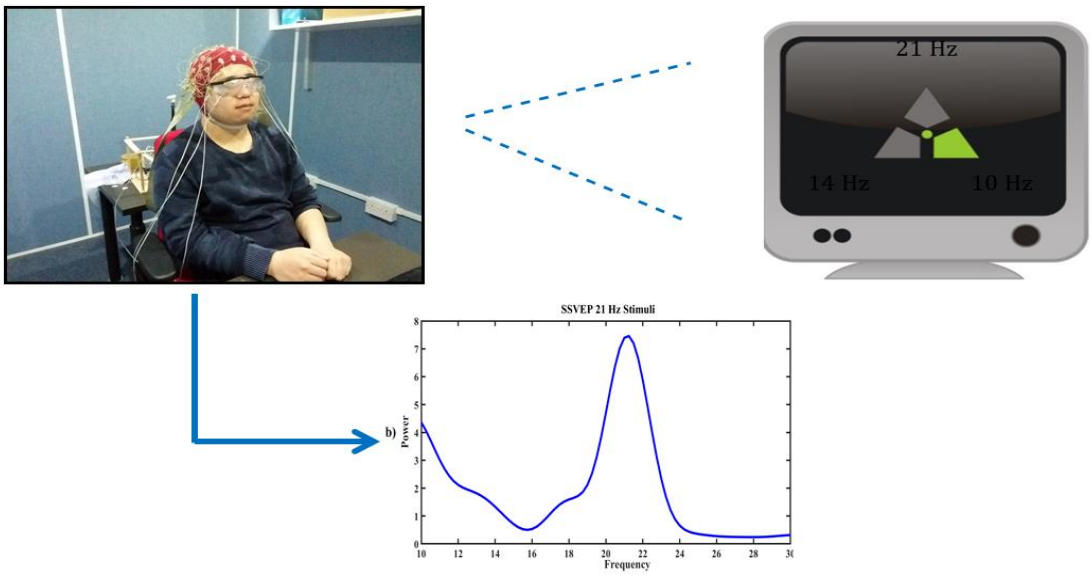

Figure 3.3 SSVEP recording experiments where a subject is looking at a display emulating flashing lights at frequencies of 10Hz, 14Hz and 21Hz, resulting in the SSVEP signals as the responses to different stimuli.

Figure 3.3 shows the process of generating the SSVEP signals as the responses to stimuli at frequencies 10Hz, 14Hz, and 21Hz during the SSVEP recording experiment.

After EEG data acquisition, the pre-processing procedure explained in Section 3.2, needs to be implemented in order to enhance the recorded signal further by removing unwanted brain-activities and preserving the relevant information.

3.2 EEG Data Pre-processing

EEG signal pre-processing needs to be conducted so that signals will be enhanced by eliminating known interference, such as artefacts and noise, while their spectral, temporal or spatial characteristics will be intensified. In this study, EEG data were analysed using EEGLAB, an open-access toolbox for EEG data analysis (Delorme and Makeig 2004).

EEGLAB is an interactive MATLAB toolbox with a collection of over 300 functions for processing continuous and event-related EEG and other electrophysiological data. It incorporates Independent Component Analysis (ICA), time/frequency analysis, artifact reduction, event related statistics and other useful modes for visualising the averaged and single-trial data. EEGLAB can run under various operating systems, i.e. Windows, Linux, Unix, and Mac OS X, and its Graphic User Interface (GUI) allows users to process their high-density EEG data flexibly and interactively (Delorme and Makeig 2004).

The pre-processing procedure includes three main steps which have been briefly explained below.

1. **Data Decimation:** Decimation is the elimination of samples in a periodic fashion (J. Wolpaw and Wolpaw 2012). When the signal is digitised at higher rate than required, for more efficient processing and less required data storage, it can be beneficial to decimate the sampled signal to a minimum effective sampling rate. In this study, EEG signals are initially recorded at the 2048 Hz sampling rate and decimated to 256 Hz in the pre-processing procedure.

2. **Filtering:** EEG signals of interest are normally in a specified range of frequencies. Therefore, eliminating the influence of the frequencies that lie outside the range of frequencies of interest is necessary. Toward this, temporal filters, including Finite Impulse Response (FIR), Infinite Impulse Response (IFR), and Discrete Fourier Transform (DFT) can be used to remove undesired effects and isolate relevant brain activities. It is worth mentioning that the width of the filter (i.e. the frequency range) should be selected carefully to prevent unnecessary loss of information. In this research, we used FIR filtering with a frequency range of 0.1Hz to 50 Hz. The Figure 3.4 shows the FIR filtering implemented on the EEG recording of one of the subjects.

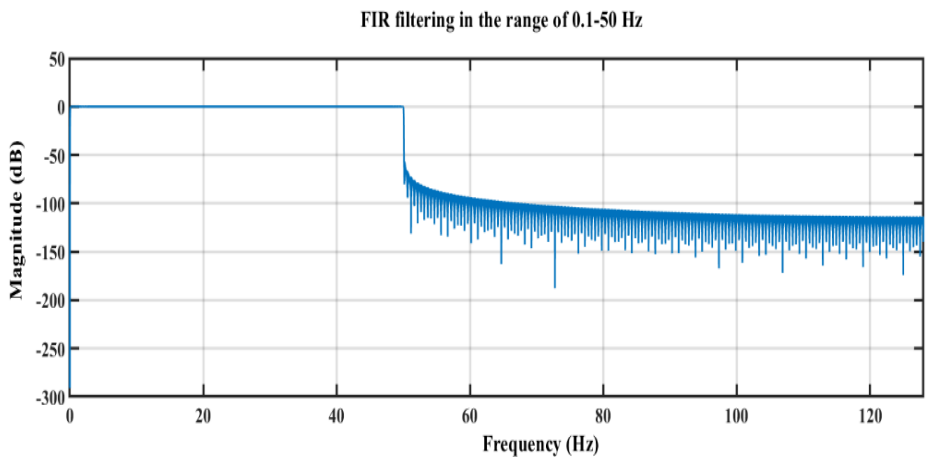

Figure 3.4 FIR filtering in the range of 0.1-50 Hz on the recorded SSVEP signal.

3. **Segmentation:** EEG signals are commonly segmented into consecutive sample blocks prior to the feature extraction process. The feature vectors are generated from the samples within each individual sample block. In this study, the recorded SSVEP signals from each subject are first separated based on the specific event using the build-in functions within the EEGLAB® toolbox. In this work, three stimuli at 10 Hz, 14 Hz, and 21 Hz represent the events within the EEGLAB® framework. The event selection will result in dividing the collected EEG signals into three separate SSVEP signals, in which each one contains the responses corresponding to a specific stimulus frequency. Then, the generated event based SSVEP signals will be segmented into segments each lasting 2.5 second for further analysis. After segmentation of the event-based SSVEP signals, removing the mean baseline value from each segment can be beneficial when the baseline differences between data segments (e.g., those arising from low-frequency drifts or artefacts) are present. These are not meaningfully interpretable, and it is useful be removed from data to avoid affecting the data analysis (Delorme and Makeig 2004). It is worth mentioning, for the recorded alpha brain activity of each subject, as they do not contain any events, they could not process in EEGLAB®. The segmentation of the alpha signals into the 2.5-second-long segments were implemented separately in MATLAB®.

4. **Artefact Reduction:** In EEG recordings, there are various types of artefacts which could arise from biological sources such as muscle and eye movements, and heartbeat activity. The movement of muscle is one the most significant artefact and very difficult to be removed or fully realised (J. Wolpaw and Wolpaw 2012). In this research, artefact reduction has been implemented using the build-in functions within the EEGLAB® toolbox. Therefore, Trials containing artefacts were removed using an automatic rejection procedure, by setting a threshold of +/-100mV. Eye movements were corrected using the Gratton-Coles procedure (Gratton, Coles, and Donchin 1983).

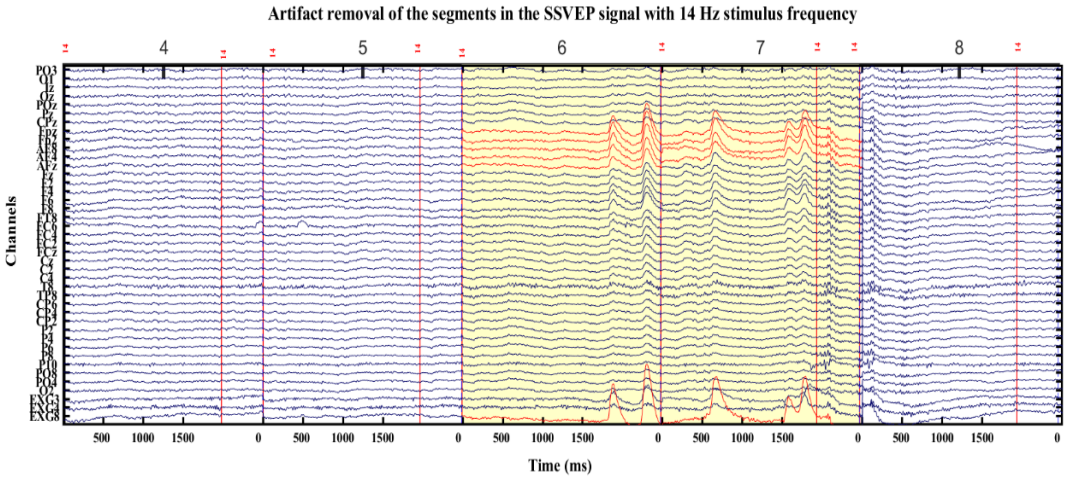

Figure 3.5 Example of artefact removal of the segments in the SSVEP signal with 14Hz stimulus frequency. Segments 6 and 7 shaded in yellow are marked for trial rejection as they contain artefact.

Figure 3.5 shows an example of artefact removal on some segments of the SSVEP signal with 14 Hz stimulus frequency. As it can be seen, segments 6 and 7 shaded in the yellow colour are marked for rejection as they contain artefacts.

After pre-processing the recorded EEG signals, the remained segments are ready for the feature extraction procedure explained in Section 3.3 to generate the feature vectors that can represent the signal characteristics in the best meaningful way. Figure 3.6 shows an example of different

segments of the recorded EEG signals from a subject in frequency-domain after data-preprocessing.

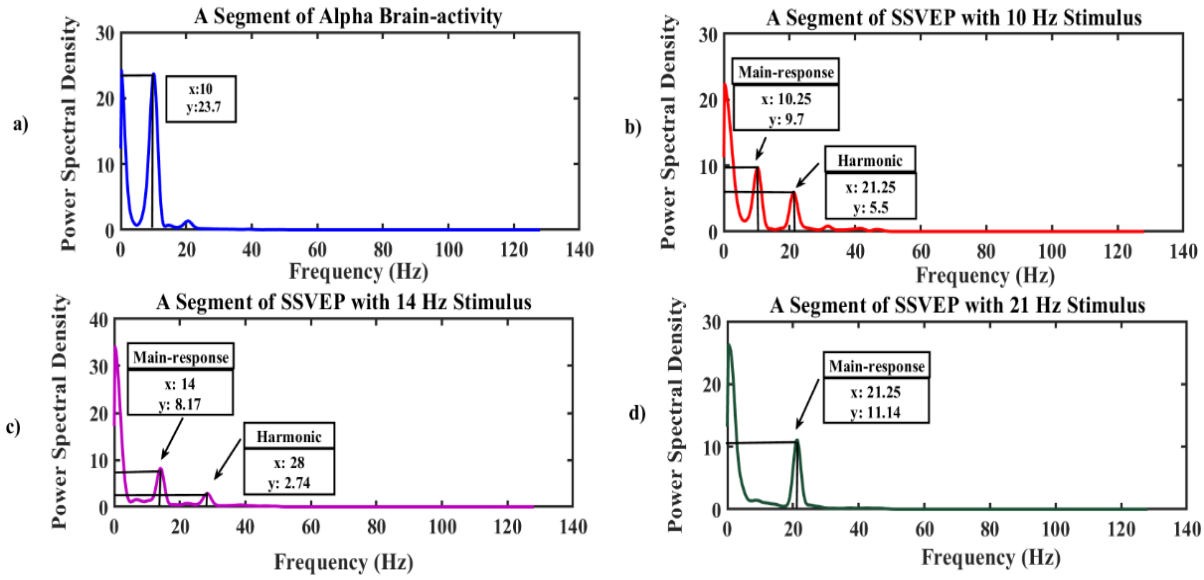

Figure 3.6 Example of segments of the recorded EEG signals from a subject in frequency-domain after data-preprocessing. a) A segment of Alpha brain-activity with relative power spectral density at 10 Hz; b) A segment of SSVEP signal with relative power spectral density at 10.25 Hz responds to 10 Hz stimulus and harmonic at 21.25 Hz; c) A segment of SSVEP signal with relative power spectral density at 14 Hz responds to 14 Hz stimulus and harmonic at 28 Hz; d) A segment of SSVEP signal with relative power spectral density at 21.25 Hz responds to 21 Hz stimulus and no harmonic.

3.3 Feature Extraction

A feature is a distinctive measurement extracted from a segment of a signal. Features are used to represent relevant characteristics of signal. Therefore, feature extraction is a vital step in EEG signal processing as selecting the relevant features has direct impact on classification

performance. In this study, discrete wavelet transforms (DWT) method has been used to obtain such representations, extracting the features of interest from the recorded SSVEP signals.

3.3.1 Discrete Wavelet Transform (DWT)

DWT method has been used in this study due to its unique properties comparing to the other approaches such as Fast Fourier Transform (FFT), Autoregressive Modelling (AR), and Short-Time Fourier Transform (STFT). DWT outperforms the aforementioned feature extraction approaches due to the following main reasons: 1) EEG signals are non-stationary and possess transient patterns. Therefore, traditional methods, such as FFT which is only suitable for extracting non-transient information in frequency-domain, is not capable of capturing the transient patterns. Furthermore, the time-domain information is not readily seen from the transformed Fourier coefficients (Adeli, Zhou, and Dadmehr 2003). Whilst, DWT is a powerful time-frequency method in capturing transient information such as sudden short-duration signal changes which are common in time-varying biomedical signals (Cvetkovic, Übeyli, and Cosic 2008). 2) FFT is not a suitable approach to extract frequency-domain information as it suffers from low-frequency resolution, as well as spectral leakage due to its windowing function. Although the AR approach is a frequency-domain method and can provide a better frequency resolution and overcome the spectral leakage problem of FFT (Al-Fahoum and Al-Fraihat 2014), AR requires proper selection of a model order. AR gives a poor spectral estimation if the model order is not selected properly. As the FFT and AR approaches provide only frequency-domain information, the STFT approach can be used to provide time and frequency localisation at the same time and overcome the spectral leakage problem of FFT by using a Hanning window. However, the fixed window size of STFT does not address the uncertainty nature of EEG signal. For example, if the window size is too narrow, it reduces the frequency

resolution and if the window size is too wide the time resolution is poor (O.A Rosso, Martin, and Plastino 2002). DWT can overcome the aforementioned problems as it can provide time-frequency localisation with a flexible window size. A wider window size is used to produce good frequency localisation at the lower frequency and a narrower window size is used for better time-localisation at the higher frequency (Mallat 1989, Jahankhani, Kodogiannis, and Revett 2006).

DWT is based on the Wavelet Transform (WT) approach which represents a time function in terms of simple and fixed blocks, known as wavelets. The building blocks are a family of functions which are derived from a single generating function called mother wavelet which involves translation and dilation operations. In WT, dilation is known as a scaling function which compress or stretches the mother wavelet and translation shifts the mother wavelet along the time axis (Cvetkovic, Übeyli, and Cosic 2008). WT is divided into two categories, namely continues (CWT) and discrete (DWT). However, in CWT, due to the continuous varying of scaling and translation parameters, the wavelet coefficients are required to be computed for every single scale. Therefore, it results in huge computational time and vast amount of data (Subha et al. 2010). In light of this, in this study, DWT approach has been selected as the feature extraction method.

Due to its multi-scale feature, DWT can decompose the signal into the number of scales. DWT computes a set of wavelet coefficients which can be used to reconstruct the original signal and provide information as a direct estimation of local energies at different decomposition scales. The scaling function $\varphi_{j,k}(n)$ and detailed function $\psi_{j,k}(n)$ used in DWT are defined in Eq. 3.1 and Eq. 3.2 respectively.

$$\varphi_{j,k}(n) = 2^{\frac{j}{2}} h\left(2^j n - k\right) \qquad (3.1)$$

$$\psi_{j,k}(n) = 2^{\frac{j}{2}} g\left(2^j n - k\right) \qquad (3.2)$$

Where $g(n)$ and $h(n)$ represent the high- and low-pass filters respectively. $n = 0, 1, 2,\ M - 1$; $j = 0, 1, 2, j - 1;\ k = 0, 1, 2, 2^{j-1}$; j=$log_2(M)$ and M is the length of the signal.

The outputs of the low- and high-pass filters are the approximation and details coefficients of the signal (Omerhodzic et al. 2013).The approximation coefficient A_i is defined in Eq. 3.3 and details coefficient D_i is computed using Eq. 3.4, with $i = 1,2,, \dots, l$ indicating the decomposition level starting from level 1 to l.

$$A_i = \frac{1}{\sqrt{M}} \sum_n x(n) \cdot \varphi_{j,k}(n) \qquad (3.3)$$

$$D_i = \frac{1}{\sqrt{M}} \sum_n x(n) \cdot \psi_{j,k}(n) \qquad (3.4)$$

The multi-scale decomposition process of an input signal $x(n)$ is shown in Figure 3.6.

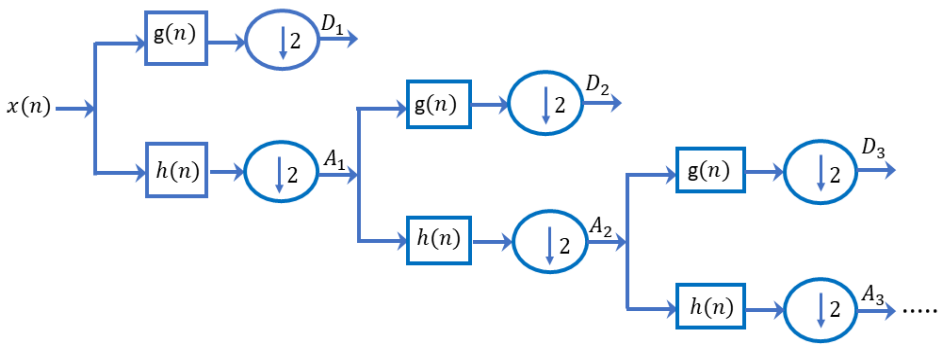

Figure 3.7 The multi-scale decomposition procedure of DWT.

As can be seen from Figure 3.7, at each stage of the decomposition scale, two digital filters and two down samplers by a factor of 2 are used. The outputs of the high- and low- pass filters are passed to down samplers of the first stage which provide the detail D_1 and approximation A_1, respectively. A_1 is further decomposed and this process continues until it reaches the specified maximum decomposition level l (Cvetkovic, Übeyli, and Cosic 2008).

In DWT, choosing the suitable wavelet and the number of decomposition level is very important for processing the input signal. The number of decomposition level is determined by the dominant frequency of the signal. Therefore, those parts of the signal that correlates well with the frequencies necessary for classification of the signal are retained in the wavelet coefficients (Subasi 2007). In this study, Daubechies 4 (db4) is selected due to its smoothing feature which is suitable for detecting changes of the EEG signal (Adeli, Zhou, and Dadmehr 2003, Omerhodzic et al. 2013). The number of decomposition levels is set to 5 in order to decompose EEG signals into five frequency bands. The frequency band of each decomposed level D_l is defined in Eq. 3.5. Where, f_s is the sampling frequency rate and l represents the decomposition level. In this work, the sampling frequency rate is 256Hz.

$$D_l = \left[{}^{f_l}/_{2}, f_l \right] \quad (3.5)$$

$$f_l = {}^{f_s}/_{2^{l+1}}$$

TABLE 3.1

FREQUENCY BANDS CORRESPONDING TO EACH DECOMPOSITION LEVELS

Levels	**Frequency range (Hz)**	**Frequency bands**
$\boldsymbol{D}_1$	(64-128)	noises
$\boldsymbol{D}_2$	(32-64)	gamma
$\boldsymbol{D}_3$	(16-32)	beta
$\boldsymbol{D}_4$	(8-16)	alpha
$\boldsymbol{D}_5$	(4-8)	theta
$\boldsymbol{A}_5$	(2-4)	delta

The decomposition levels and corresponding frequency bands are shown in TABLE 3.1. The signals are decomposed into five details ($D_1 - D_5$) and one approximation (A_5).

Figure 3.8 shows the 5-level decomposition process of a SSVEP segment with 14 Hz stimulus.

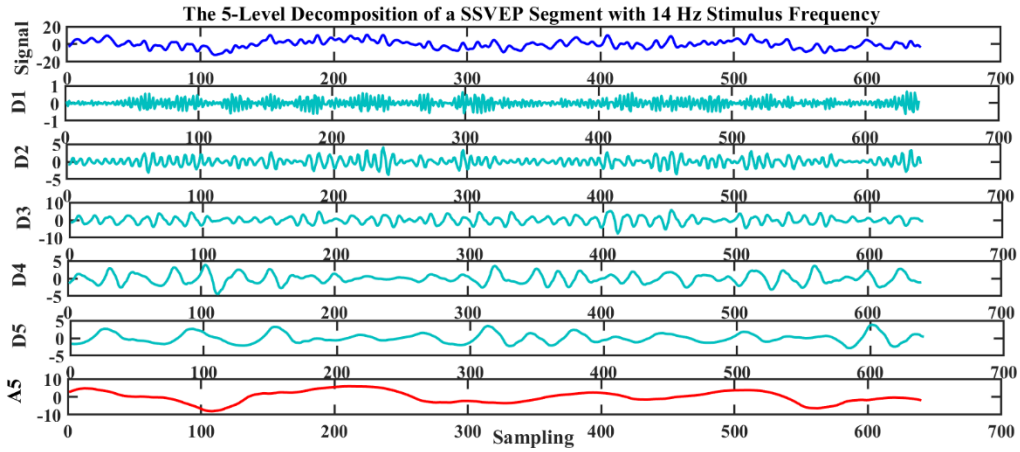

Figure 3.8 5-level decomposition of SSVEP signal using DWT. The segment is decomposed into five details ($D_1 - D_5$) and one approximation (A_5).

3.3.2 Feature Vector Preparation

A feature vector consists of a set of all features that represents the signal with the least minimum loss of information (Cvetkovic, Übeyli, and Cosic 2008). In this study, SSVEP signals with three different stimulation frequencies at 10Hz, 14Hz and 21Hz, and the baseline signals with a peak at 10Hz from 10 subjects are used in the feature extraction procedure using the DWT method described in Section 3.3.1. It is also worth mentioning that in this study, the features are extracted from only O1 channel of the ten subjects. It is believed that SSVEPs have higher amplitudes at O1, O2 and OZ channels (Pastor et al. 2003).

Based on the DWT method, discrete wavelet coefficients for each SSVEP recording under different stimulation frequencies from ten subjects have been computed based on Eq. 3.4 for three sub-bands, i.e. gamma (D_2), beta (D_3), and alpha (D_4). The selections of three sub-bands is based on the stimulation frequencies of SSVEP signals and their harmonics, which are located in these sub-bands. The discrete wavelet coefficients for the recorded baseline signals from every subject have also been computed for the same three bands.

Three main features characterised by relative wavelet energy, Shanon wavelet entropy, and the mean power of details coefficients are computed for the three sub-bands based on their obtained wavelet coefficients. The reason to create the feature vectors based on these three features instead of using the computed wavelet coefficients directly is to reduce the dimensionality of the feature vectors and better represent the time-frequency distribution of the recorded signals.

A brief deception of these three features are given in below:

1. Relative Wavelet Energy: The energy of wavelet coefficients at each decomposition level represents the strength of the signal. The wavelet approximation coefficient (A_i) and detail coefficients (D_i) of a signal at the *ith* level are obtained based on Eq. 3.3, and Eq. 3.4, respectively. Therefore, the energy of detail and approximation are computed using Eq. 3.6. and Eq. 3.7. respectively. Where, E_{Di} represents the wavelet energy of detailed coefficients; E_{Ai} represents the wavelet energy of approximation coefficients at the *ith*

   Level; and *N*, represents the number of details coefficients.

$$E_{Di} = \sum_{j=1}^{N} |D_{ij}|, \quad i = 1,2, \dots . l \qquad (3.6)$$

$$E_{Ai} = \sum_{j=1}^{N} |A_{ij}|, \quad i = l \qquad (3.7)$$

   The total energy is calculated using Eq. 3.8, where,E_T represents the total energy of the signal.

$$E_T = (\sum_{I=1}^{l} E_{Di} + E_{Al}) \qquad (3.8)$$

   Therefore, the relative energy (E_{Rengi}) at the ith level can be obtained using Eq. 3.9. Relative energy provides information regarding the energy for the specific frequency band and gives a degree of similarity between segments of a signal (Garg et al. 2011).

$$E_{Rengi} = \frac{E_{Di}}{E_T} \qquad (3.9)$$

2. Shanon Wavelet Entropy: Wavelet entropy is based on Shanon entropy (Shannon 1948), Shanon wavelet entropy gives a degree of order/disorder of the signal (Osvaldo A. Rosso et al. 2001). Shanon wavelet entropy (SW_i) at the ith level is computed using Eq. 3.10.

$$SW_i = -(\sum E_{Rengi} \cdot \log E_{Rengi}) \qquad (3.10)$$

3. Mean Power of Details Coefficients: Mean power of details coefficients ($meanp_i$) represents the local wavelet power spectrum at the ith level (specific frequency band) which provides a degree of cumulative information of variation at each level of decomposition (Subramani, Sahu, and Verma 2006). $meanp_i$ is computed using Eq. 3.11.

$$meanp_i = {}^{1}/_{Ni} \left(\sum_{j=1}^{N} D_{ij}^2\right) \quad (3.11)$$

TABLE 3.2

DATASETS PREPARED FOR EACH SUBJECT BASED ON THE EXTRACTED FEATURES.

Subjects	SSVEP 10 Hz	SSVEP 14 Hz	SSVEP 21 Hz	Baseline 10Hz
S1	105×9	125×9	134×9	59×9
S2	159×9	156×9	143×9	59×9
S3	267×9	297×9	303×9	63×9
S4	133×9	104×9	132×9	59×9
S5	233×9	374×9	370×9	59×9
S6	268×9	166×9	154×9	59×9
S7	258×9	269×9	350×9	59×9
S8	51×9	47×9	60×9	59×9
S9	83×9	80×9	105×9	59×9
S10	128×9	131×9	232×9	59×9

Therefore, for each subject and the recorded signal, a dataset consisting of all features is represented as a matrix of size $S \times N$ where, S represents the number of segments (i.e. the number of data points in the dataset) in a specific signal and N represents the total of nine

extracted features for each segment. The datasets prepared for ten subjects are summarised in TABLE 3.2.

3.4 Conclusion

In this chapter, the full procedure of the EEG-data acquisition has been explained. The pre-processing data procedure for enhancing the recorded signals prior to the feature extraction has been presented. Finally, the feature extraction procedure based on the DWT method has been explained in which the feature vectors are generated and prepared for the classification algorithm presented in chapter 4.

Chapter 4

Classification of Multi-class EEG data using Immune Multi-Objective Fuzzy Modelling

In this chapter, an Immune Inspired Multi-Objective Fuzzy Modelling (IMOFM) mechanism, which was originally designed for regression problems, is adapted for classification of multi-class EEG data. IMOFM features a multi-stage modelling procedure combining a gradient based local search and a metaheuristic based multi-objective search algorithm. Due to this multi-stage modelling procedure, IMOFM leads to not only a high prediction accuracy but also a simplified model structure.

Simplification of the model structure is carried out by removing redundant (unnecessary) components within the model and this generally leads to a better generalisation capability and improved interpretability of the model. Two main contributions of this chapter are as follows:

1. The multi-objective modelling framework of IMOFM has been exploited to encourage the diversity of the elicited models which can be used in designing effective ensemble classifiers in Chapter 5. There are several diversity measures which are closely linked to either parameter or structure optimisation procedures of IMOFM. These diversity measures can be used as one of the objective functions to elicit the desired FRBSs. However, choosing the most appropriate diversity measure is a nontrivial task, as it pertains to the inherent mechanisms of IMOFM and lays down the foundation for designing effective ensemble classifiers in the next chapter.

2. A decomposition-based approach is employed to classify multi-class EEG data. There are two main approaches to deal with multi-class problems, namely the decomposition-based method and All-At-Once (AAO) method. The decomposition-based method is based on creating binary classifiers for different pairs of EEG signals. Each pair of EEG signals is generated using a different pair of stimuli. The decomposition-based method generally leads to high classification accuracy and is therefore the choice of this project. Based on the binary classifiers, different aggregation approaches, including variants of majority voting and Decision Directed Acyclic Graph, are proposed and investigated. Comparing to various baseline classification algorithms, such as the adaptive neuro fuzzy inference system (ANFIS), artificial neural networks (ANNs), and the CART decision tree algorithm, the proposed IMOFM based multi-class classification framework presents superior performance in terms of both accuracy and interpretability.

In the following, we first introduce the multi-class classification problem. The proposed binary classification mechanism using IMOFM is then discussed along with the choice of the objective functions and their impact on accuracy, diversity and interpretability of the elicited classifiers. Building on the binary classifiers, the aggregation approaches and the proposed multi-class classification framework are then discussed. Finally, the experimental and comparison results are presented.

4.1 Multi-class Classification

As discussed in Chapter 3, there are four different classes including Class 1 for Alpha brain activities, Class 2 for the SSVEP signal with the 10Hz stimulus, Class 3 for the SSVEP signal with the 14Hz stimulus and Class 4 for SSVEP signal with the 21Hz stimulus.

It is shown that for the multi-class classification problem, decomposition-based methods can simplify the decision making as they distinguish between only two classes rather than dealing with more than two classes at the same time (Galar et al. 2011). Following this line of methods, a multi-class problem is first decomposed into several easier-to-solve binary classification problems. Unlike decomposition-based approaches, in All-At-Once (AAO) approaches, the decision has to be made with respect to all the classes at once, leading to a significant increase in the computational cost comparing to decomposition-based approaches (Tsujinishi, Koshiba, and Abe 2004). It is also worth mentioning that, the generalisation capability of AAO approaches have been reported to be poorer than decomposition-based approaches, particularly in comparison to one type of decomposition-based approaches, i.e. the One-Against-One (OAO) approach which will be discussed below (Tsujinishi, Koshiba, and Abe 2004). In light of the above reasons, decomposition-based approaches have been chosen in this work for classification of multi-class EEG data.

There are two main decomposition-based approaches: One-Against-One (OAO) and One-Against-All (OAA) approaches. Both OAO and OAA approaches decompose a multi-class problem into several binary classification problems. In this work, OAO is used due to its generalisation capability. Following the OAO approach, for the multi-class EEG data we have in this work, six different binary classifiers have to be trained for each subject in order to cover the complete combination of different classes. These classifiers include: CF1 to classify between Classes 1 and 2, CF2 to classify between Classes 1 and 3, CF3 to classify between Classes 1 and 4, CF4 to classify between Classes 2 and 3, CF5 to classify between Classes 2 and 4, and CF6 to classify between Classes 3 and 4.

4.1.1 Decomposition based Methods

Classification tasks have been applied to different real-world applications, many of which involve multi-class classification, e.g., sign language recognition tasks (Aran and Akarun 2010), classification of EEG signals (Guler and Ubeyli 2007) and medical applications such as cancer diagnostics (Anand and Suganthan 2009). Decomposition based methods divide the multi-class problem into a series of binary classification problems that simplify the decision boundaries of the multi-class problem (Galar et al. 2011). The two most well-known methods within this category are OAA and OAO. In the OAO approach, it divides the multi-class problem into binary classes which include all the possible combinations of pairwise classes. Therefore, the OAO approach breaks down the multi-class problem into distinguishing classes within each pair of classes. Similar to the OAO approach, the OAA approach also divides the multi-class problem into binary classes. However, each binary classifier needs to distinguish one of the classes from all other classes (Rifkin and Klautau 2004). In the learning phase of the OAO approach, the training set for each binary classifier only includes the samples from the corresponding two classes. Therefore, OAO requires less training time. On the contrary, training a binary classifier based on OAA requires that the samples are from all classes, of which one will be considered as positive and all other classes negative. Therefore, training an OAA based classifier is computationally more expensive than using the OAO approach. It is also worth mentioning that having a training dataset consisting of samples from either a single class or all other classes will produce an imbalanced dataset, leading to some undesirable effects in the trained classifiers (Galar et al. 2011, Eichelberger and Sheng 2013).

Both of the above decomposition-based methods require an extra step to combine or aggregate the results from the binary classifiers in order to provide the final decision. A brief review of

aggregation techniques, in particular those for the OAO approach, are provided in the next section.

4.1.2 Aggregation Methods

Given a classification problem, if N represents the number of classes, a training data set is denoted as $\{(X_1, y_1), \dots (X_i, y_i) \dots (X_D, y_D)\}$, where, $X_i \in R^m$ represents the feature vector of the ith data point, D is the number of data points in the training data set, and $y_i = \{1, \dots \dots N\}$, is the class label of the ith data point. Therefore, a multi-class classification task is to construct a mapping $F: X \rightarrow \{1, \dots N\}$ using the labelled training data set through the training process. As the OAO method is used in this work, the focus in this section is placed on the aggregation methods for OAO.

1. Voting Strategy: In this approach, each binary classifier contributes to a vote for the predicted class. The summation of the votes for each class will be computed. The class with the maximum votes will be the predicted class for the testing sample.

2. Averaged voting strategy: In this approach, each binary classifier contributes to a vote for the predicted class in the same way as what the voting strategy does. The votes for each classifier will be added and averaged and the averaged vote will be considered as the final label.

3. Decision Directed Acyclic Graph (DDAG): DDAG is based on the Directed Acyclic Graph, which is an oriented graph with no cycles (Platt, Cristianini, and Shawe-Taylor 1999). DDAG's structure is based on the rooted binary tree with the maximum 2 children. In a classification task based on DDAG, e.g. there are $N = 4$, classes, therefore, there are 6 binary OAO based classifiers, one for each pair of classes, hence there exists $K = 6$

internal nodes. The prediction y, for a sample $X \in R^m$, is determined by a binary classifier with a mapping function of $f_{ij}(X)$, as defined in Eq. 4.1 (Zhou, Wang, and Fujita 2017), where $c$mapping is a constant, and i and j represent two different classes.

$$y = \begin{cases} i, & f_{ij}(x) \geq c \\ j, & otherwise \end{cases} \tag{4.1}$$

An example of using DDAG for aggregating OAO based binary classifiers with $N = 4$, i.e. 4 classes, is shown in Figure 4.1.

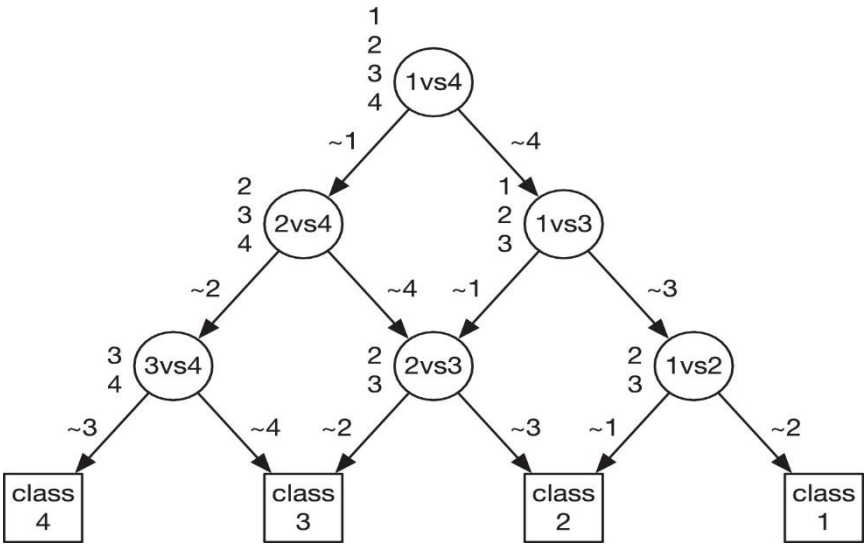

Figure 4.1 An example of using DDAG for a multi-class classification problem with N = 4.

In this example, the classification of a sample X starts at the root node which represents a binary classifier with the mapping function of $f_{1,4}(X)$. If $f_{1,4}(X) \geq c$, X does not belong to class 4 and the node exits the right edge to the next node with a binary classifier with the mapping function of $f_{1,3}(X)$. Similarly, if $f_{1,4}(X) < c$, x does not belong to class 1 and the root node exits the left leg to the next node where the mapping function of $f_{2,4}(X)$ will be evaluated. The above process repeats until it reaches the final leaf node. The path taken

through DDAG is known as the evaluation path (Platt, Cristianini, and Shawe-Taylor 1999). The mapping functions in DDAG are carried out based on a list. At the root node, in the beginning of the classification, the list is initialised containing all the classes. In the example shown in Figure 4.1, at the root node, the initial list $l = \{1,2,3,4\}$ contains four classes. The binary classifier at the root node distinguishes class 1 vs class 4, which corresponds to the first and last classes in the initial list. If the evaluation of X using $f_{1,4}(X)$ indicates that X belongs to class 1, then class 4 will be eliminated from the list and the updated list will be $l = \{1,2,3\}$ for the next node. The binary classifier with the mapping function of $f_{1,3}(X)$ at this next node distinguishes between the first and last classes within the updated list $l = \{1,2,3\}$, i.e. class 1 vs class 3. If the predicted class for x is not class 3, the list will be updated for the next node by eliminating class 3, i.e. $l = \{1,2\}$. At this new node, the binary classifier with the mapping function of $f_{1,2}(X)$ distinguishes between the two remaining classes, i.e. class 1 vs class 2. If x does not belong to class 2, class 2 will be removed from the list. The only remaining class in the final list is class 1, which is the final predicted class for X. The termination condition for DDAG is when there is only one class in the list. Therefore, with N classes, $N - 1$ nodes will be evaluated in order to predict the final class label.

It is worth mentioning that the list in DDAG is only used to update the remaining class labels at each level of the evaluation path. These class labels in the list form a pool of candidate classes based on which the mapping function (i.e. the binary classifier) will be chosen. In DDAG, choosing the binary classifier at each node using the first and last classes in the list is only making sense when these two classes are the most distinguishable classes from each other within the list. In light of this, in (Takahashi, Takahashi, and Abe 2003), at each node, the binary classifier with the minimum classification error for the training dataset is selected as long as the two classes of this binary classifier are included the list. If

X does not belong to one of the classes that class will be eliminated from the list for the next node.

In this work, several variants based on (Takahashi, Takahashi, and Abe 2003) are proposed to select the binary classifier for each node according to the list. The details of these variants and results will be presented in Sections 4.3 and 4.4.4 respectively.

4.2 Binary Classification based on IMOFM

4.2.1 IMOFM

In this section, Immune Inspired Multi-Objective Fuzzy Modelling (IMOFM) algorithm, which was originally proposed for regression problems (J. Chen and Mahfouf 2012), is first adapted for binary classification of EEG data. IMOFM is a metaheuristic based multi-objective fuzzy modelling approach which offers high prediction accuracy as well as a simple/interpretable model structure. IMOFM utilises a multi-objective metaheuristic to remove the redundant (unnecessary) components of the predictive model, leading to better generalisation capability and interpretability of the model. Like ANNs and other nonlinear classification algorithms, Fuzzy Rule based Systems (FRBSs) are the universal approximator (Benítez, Castro, and Requena 1997), that is fit for the classification task in this work. However, there are several distinct features that justify the choice of IMOFM as the classification algorithm in this work rather than other machine learning based classification algorithms. (1) IMOFM employs interpretable FRBSs to represent the system under investigation (J. Chen and Mahfouf 2012). Moreover, through the simplification stage, the generalisation capability of FRBSs is improved, effectively avoiding the overfitting problem. (2) FRBSs, in general, provide a good platform to deal with noisy, imprecise or incomplete information which is often presented in

many human-cognition systems (Cordón, del Jesus, and Herrera 1999). FRBSs allow modelling the uncertainty associated with input measurements, as well as with providing the relationship between the input features and output classes (Pota, Esposito, and De Pietro 2017). (3) IMOFM, through multi-objective search, encourages diversity of the elicited FRBSs, paving the foundation to develop ensemble fuzzy classifiers (see Chapter 5), which can further improve the classification accuracy. In the following, the key steps involved in IMOFM are briefly reviewed.

IMOFM involves three stages, in which the first two stages aim to generate an initial accurate approximate FRBS. In the first stage, an evolutionary based *K*-means clustering algorithm is used to extract the FRBS from the dataset. This initial FRBS can be further improved as 1) it contains redundant fuzzy sets and rules; therefore, the elicited FRBS offers low interpretability; and 2) the parameters of the FRBS can be further tuned to give more accurate predictions. With respect to the latter point, a constrained Back-Error-Propagation (BEP) algorithm is used in the second stage to improve prediction accuracy of the initial FRBS. With respect to the first point, the improved FRBS will be used to seed the initial population in the third stage. Since, the membership function parameters and the rule-based structure cannot be simultaneously optimised in Stages 1 and 2, only a sub-optimal FRBS can be obtained. Therefore, in the third stage, a multi-objective optimisation problem can be formulated to simultaneously refine the parameters of the membership functions as well as the structure of FRBSs. The choice of different objectives will be explained in Section 4.2.2.

It is worth mentioning that in the third stage, IMOFM applies simplification procedures to remove redundancy both in the rules and fuzzy sets. The simplification steps include removing unimportant rules, merging similar rules, removing universal fuzzy sets, and merging similar fuzzy sets. More details for the aforementioned steps can be found in (J. Chen and Mahfouf 2012).

As mentioned earlier, IMOFM is originally designed for regression problems. Therefore, the defuzzified output of a Mamdani FRBS will go through a post-processing procedure so that the defuzzified output will be rounded up or down to either 1 or 0 in this work, representing two different class labels in our classification problems.

4.2.2 Interpretability of FRBSs

Interpretability of FRBSs can be defined as understandability or transparency of the generated rules (Yaochu Jin 2000). The interpretability of FRBSs depends on several aspects (Y. Jin 2000, Ishibuchi and Yamamoto 2004). (1) Comprehensibility of fuzzy sets. In order to have understandable fuzzy rules, fuzzy sets should be distinguishable so that a meaningful linguistic term can be associated with them. (2) Simplicity of FRBSs, which means the number of input variables involved in each rule and the total number of fuzzy if-then rules should not be excessive. (3) The fuzzy rules in the rule base should be consistent. If there are rules which are significantly contradictory to each other, it is difficult to understand the rules and it will confuse the users. In IMOFM, the interpretability of FRBSs has been achieved through (1) simplifying the rule-based structure, and (2) refining the membership function parameters of FRBSs.

The interpretability aspect of FRBSs is one of the essential requirements in a modelling process, with another one being the accuracy of the model. However, these two requirements cannot always be processed at the same time and a good balance between them is the best outcome that one can achieve (Jun 2010). In general, reducing the complexity of the model structure will adversely affect the prediction accuracy of the model. Therefore, the accuracy vs interpretability issue of fuzzy modelling can be formulated as a multi-objective optimisation problem, in which the predictive error and the complexity of the FRBS are minimised simultaneously.

The interpretability aspects as implemented in IMOFM are closely related to the diversity concept, in particular the structural diversity (Sharkey 1999, Brown and Brown 2004 , Masisi, Nelwamondo, and Marwala 2008). Therefore, several diversity measures that are widely used in constructing ensemble neural networks (Liu and Yao 1999, Chandra and Yao 2004, Chandra and Yao 2006) are discussed in the next section. We will investigate in the next section if these diversity measures are fit as part of the objective functions comparing to the interpretability measure.

4.2.3 The Objective Functions

The decision on formulating a multi-objective optimisation problem is depending on the ultimate aim of designing a well-performed classification algorithm. Therefore, the accuracy of prediction should be one of the vital objectives. As mentioned in the previous section, we treat the classification problem as a regression problem initially. After the rounding up/down of the defuzzified output, a class will then be assigned to the data sample under investigation. Therefore, the accuracy of prediction can be defined in Eq. 4.2, which is to minimise the root mean squared error (RMSE) (J. Chen and Mahfouf 2012).

$$Prediction\ error = \sqrt{\frac{\sum_{i=1}^{N}\left(y_i^{predict} - y_i^{real}\right)^2}{N}} \quad (4.2)$$

As it can be seen from Eq. 4.2, the accuracy of FRBSs can be improved by minimising the RMSE between the predicted value $y_i^{predict}$ and the real class label y_i^{real} for the training sample X_i..

Another commonly used objectives for designing a classification mechanism based on FRBSs is the interpretability of FRBSs. Ishibuchi, in designing pattern classification algorithms

(Ishibuchi, Nakashima, and Murata 2001) and later on in designing a fuzzy ensemble classifier (Ishibuchi and Nojima 2006) based on FRBSs, formulated a combination of three objectives, including maximising the number of correctly classified patterns, minimising the number of fuzzy rules and minimising the total number of antecedents. The first objective concerns the classification accuracy similar to the RMSE defined in Eq.4.2. The last two are related to the interpretability of FRBSs. The interpretability of FRBSs in this work is defined in Eq.4.3. By reducing the redundancy in the FRBS structure, the complexity of FRBS will be minimised, leading to improved interpretability of the obtained FRBSs (J. Chen and Mahfouf 2012).

$$Complexity = NumR + Numset + RL \qquad (4.3)$$

Where, NumR is the number of rules. Numset is the number of fuzzy sets. RL is the summation of the rule length of each rule.

Diversity is another commonly used objective for designing an ensemble of classifiers (Kuncheva and Whitaker 2003). By evolving as diverse as possible and accurate enough classifiers through a multi-objective optimisation procedure, a set of classifiers can be generated. By combining these classifiers, the obtained ensemble model often outperforms each individual classifier within such a set. In (Chandra and Yao 2004), an attempt to design an accurate yet diverse ensemble of neural networks has been made by formulating a multi-objective optimisation problem with respect to diversity and accuracy as the two objective functions. The diversity measure in (Chandra and Yao 2004) is based on Negative Correlation Learning (NCL). The idea behind NCL is to encourage different individual classifiers in an ensemble to learn different parts of training dataset so that the ensemble can perform the classification task on the whole data set better than the individual classifier (Y. Liu and Yao 1999). Apart from using NCL as a separate objective function, NCL has also been used as a penalty function. Minimising NCL is equivalent to minimise mutual information between the outputs of the classifiers, hence making them different and complement to each other (Chandra

and Yao 2006). Therefore, the lower the NCL value is, the higher the diversity of the classifiers is in an ensemble. NCL is defined in Eq. 4.4.

$$NCL_{cl} = \sum_{i=1}^{Num}(f_{cl}^{i} - f^{i})\left[\sum_{j=1,j\neq cl}^{M}\left(f_{j}^{i} - f^{i}\right)\right] \qquad (4.4)$$

In Eq. 4.4, Num is the number of data points in the training set; M is the number of classifiers in the ensemble; for each individual classifier cl, Eq. 4.2 indicates how different classifier cl is from other classifiers in the ensemble. f^{i} represents the ensemble output for training sample i.

In (Chandra and Yao 2006), a different measure of diversity known as Pairwise Failure Crediting (PFC) has been proposed to design accurate, yet diverse ensemble of neural networks. PFC can also be used as a penalty function. However, unlike NCL that compares the output of a classifier only to the output of the ensemble, PFC measures the error of individual classifier on the training set with respect to all other classifiers in the ensemble. PFC has the maximum value of one and minimum value of zero. A classifier with the PFC value of one means it is very different from all other members in the ensemble. While a classifier with the PFC value of zero means that all classifiers in the ensemble are similar and they all have zeros as their PFC values. Therefore, in formulating a multi-objective optimisation problem, using PFC as one of the objectives is to maximise PFC (Chandra and Yao 2006). PFC is computed for each individual classifier k, as defined in Eq. 4.5 (Bhowan et al. 2013).

$$PFC_{cl} = \frac{1}{M-1}\sum_{j=1,j\neq cl}^{M}\frac{\sum_{i=1}^{Num} I\left(la_{i}^{cl}, la_{i}^{j}\right)}{Err^{cl} + Err^{j}} \qquad (4.5)$$

Where, $$I\left(la_{i}^{cl}, la_{i}^{j}\right) = \begin{cases} 1 & if\ prd\left(la_{i}^{cl}\right) \neq prd\left(la_{i}^{j}\right) \\ 0 & otherwise \end{cases}$$

In Eq. 4.5, Num is the number of data points in the training set, and M is the number of classifiers in the ensemble. la_{i}^{cl} and la_{i}^{j} represent the predicted class labels for training sample i. Indicator function $I(\cdot)$ returns 1 if the predicted class labels between two classifiers are

different for a given input, or 0 otherwise. The errors, Err^{cl} and Err^{j}, are the number of incorrect predictions for classifiers cl and j in the ensemble.

In (Bhowan et al. 2013), the performance of the ensemble classifier based on Genetic Programming (GP) using PFC and NCL respectively as one of the objectives for classification of unbalanced datasets were compared. The results showed that PFC is more effective in evolving diverse classifiers comparing to NCL.

As will be discussed in the next chapter, our aim is to design a well-performed ensemble of classifiers by generating diverse yet accurate classifiers. Therefore, it is important to investigate if these two diversity measures should be included in IMOFM. As discussed earlier, NCL and PFC are two explicit diversity measures that can be used to directly encourage the difference among the evolved classifiers. This can be achieved through optimising the parameters and/or structures of the classifiers. From this perspective, it is worth mentioning that the interpretability measure mentioned earlier in this chapter also implicitly promotes diversity through encouraging a set of FRBSs with varied structures.

In light of the above discussion, along with RMSE, either NCL, PFC, the interpretability measure, or a combination of them are justifiable candidate objective functions to be incorporated in IMOFM. Therefore, it is vital to investigate the right combination of these objectives in formulating the multi-objective classification problem to elicit diverse yet accurate classifiers in the remaining part of this chapter. In TABLE 4.1, we summarise the main features of these potential objective functions.

TABLE 4.1

COMMONLY USED OBJECTIVES IN DESIGNING A CLASSIFICATION ALGORITHM

Objectives	Aims	Approaches
Accuracy	Improve the classification accuracy	Minimisation of the prediction error, e.g. RMSE
Interpretability	Improve the interpretability of FRBSs	Minimisation of the complexity of FRBS structure
Negative correlation Learning (NCL)	Improve the diversity among the evolved classifiers	Minimisation of mutual information among the members of the ensemble
Pairwise Crediting Failure (PFC)	Improve the diversity among the evolved classifiers	Maximisation of irrelevancy among the members of the ensemble

4.3 Multi-class Classification using IMOFM based Binary Classifiers

As discussed in Section 4.1.2, in this work, the aggregation method based on the DDAG approach is used. In (Takahashi, Takahashi, and Abe 2003) for classification of a data point in a testing dataset, a classifier with the least training classification error will be chosen from the list. One of the drawbacks of this approach is that the binary classifier with the minimum training classification error is always chosen at each level of the evaluation path. The choice of this classifier is regardless of the testing data point. Therefore, a testing data point that does not belong to any of the two classes pertaining to that classifier will still be classified as one of the two classes, resulting in the wrong classification from the very beginning.

In light of the above problem, an improved DDAG approach is proposed in this work utilising the relationship the testing data point and binary classifiers. This improved DDAG is termed as DDAG-Distance. In DDAG-Distance, for a binary classifier, the Euclidean distance of a testing data point to the centres of each rule is calculated. Among the calculated distances between the data point and the centres of each rules, the minimum one is used to represent the

distance of this testing data point to the binary classifier. Therefore, at each node of the DDAG-Distance tree, the classifier with the minimum Euclidean distance to the testing data point will be chosen from the list. Compared to the DDAG approach in (Takahashi, Takahashi, and Abe 2003), the proposed DDAG_Distance approach takes each testing data point into account with the aim to assign the most suitable classifier at each node for each specific testing data point.

We have applied the DDAG-Distance approach to aggregate the IMOFM based binary classifiers and the results are reported in Section 4.4.4.

4.4 Experiments

Various experiments have been carried out to investigate: 1) the best combination of potential objective functions that work better with IMOFM; the combination of objective functions which leads to the FRBSs of the highest accuracy, diversity and better interpretability will be chosen as the objective functions used in IMOFM; 2) parameter analysis to identify the sufficient number of generations so that the IMOFM classification mechanism can be fully converged, and the optimal combination of minimum and maximum number of rules in the IMOFM classification mechanism; and 3) the performance of the IMOFM classification algorithm compared to the three baseline algorithms including Artificial Neural Network (ANNs), Adaptive Neuro Fuzzy Inference System (ANFIS) and CART decision tree algorithm, on both binary and multi-class classification problems.

4.4.1 Datasets

The datasets that have been used for the above experiments include two benchmark datasets and the recorded EEG datasets. Without loss of generality, the results based on the EEG datasets of 4 subjects are randomly chosen from 10 subjects, including two males and two

females. There are no particular preferences in choosing these 4 subjects. The results reported here are consistent with those based on other subjects.

The details of EEG data recording, EEG data pre-processing and feature extraction have been discussed in Chapter 3. The benchmark datasets include Australian credit card assessment dataset and Diabetes dataset, which are freely available on the UCI Machine Learning Repository. Tables 4.2 and 4.3 summarise the properties of the benchmark and EEG datasets respectively. The data sets are partitioned into 75% for training and 25% for testing.

TABLE 4.2

PROPERTIES OF BENCHMARK DATASETS

Benchmark Dataset	Number of Instances	Number of Features	Class Distribution
Australian Credit Card Dataset	690	14	307/383
Diabetes Dataset	768	8	268/500

TABLE 4.3

PROPERTIES OF EEG RECORDING DATASETS

EEG datasets	Subject 1	Subject 2	Subject 3	Subject 4
No. Instances & Class distribute of CF1	164 59/105	218 59/159	330 63/267	192 59/133
No. Instances & Class distribute of CF2	184 59/125	215 59/156	360 63/297	163 59/104
No. Instances & Class distribute of CF3	193 59/134	202 59/143	366 63/303	191 59/132
No. Instances & Class distribute of CF4	230 105/125	315 159/156	564 267/297	237 133/104
No. Instances & Class distribute of CF5	239 105/134	302 159/143	570 267/303	265 133/132
No. Instances & Class distribute of CF6	259 125/134	299 156/143	600 297/303	236 104/132
Number of Features	9	9	9	9

The descriptions of the above datasets are given below.

1. **The Australian credit card assessment.** This benchmark dataset has 14 attributes which are used in the evaluation of credit card applications. The dataset has 690 instances with 14 features and 1 class label (Chandra and Yao 2004). Out of the 14 features, six are continuous and eight are discrete. The mix of continuous and discrete features have made this dataset challenging. Furthermore, this dataset contains missing values which have been replaced either by the mode of the feature in case of district features or the mean of the feature in case of continuous features (Chandra and Yao 2004). The class label is binary, where 1 presents that the credit card is awarded and 0 otherwise. The class distribution in this dataset is quite balanced, where 307 instances belong to one of the classes and 383 instances belong to the other one.

2. **The diabetes dataset**. This benchmark dataset contains 8 features which are used to examine if a patient is diagnosed with diabetes or not. This dataset has 768 instances, each of which consists of 8 features and 1 class label. In this dataset all attributes are continuous. The class label is binary, where one indicates the positive test and zero otherwise. The class distribution in this dataset is unbalanced with 500 instances belong to one class and 268 instances belong to the other one, which makes this dataset a challenging classification task (Chandra and Yao 2004).

3. **The EEG recorded datasets.** The EEG datasets are recorded from a 21-year-old female subject, a 24-year-old male subject, a 24-year-old female subject and a 27-year-old male subject. These datasets are multi-class data with 4 different classes. Therefore, for each subject, there are six binary classifiers (CF1, CF2, CF3, CF4, CF5, CF6) corresponding to six different datasets. Each of these datasets contains 9 features and 1 class label. All features are continuous, and the class label is binary. The features include relative wavelet energy, Shanon wavelet entropy and mean power. Each of these features are calculated for three different frequency bands (gamma, beta, alpha). Therefore, in total, each dataset

contains 9 features. More details of feature extraction for the recorded EEG data can be found in Chapter 3.

4.4.2 Performance Metrics

The following performance metrics are used to carry out the parameter analysis and evaluate the performance of different classification algorithms. The first category is concerning the parameters of the IMOFM mechanism which includes Inverted Generalisation Distance (IGD) and Hypervolume (HV). The second category measures the classification performance, which include sensitivity, specificity, accuracy and the number of misclassified data points by a classifier.

A brief description of the above metrics is given below.

Inverted Generational Distance (IGD). In multi-objective optimisation problem, an approximation set $A = \{s_1, s_{2,} \ldots\ldots, s_{|A|}\}$ is evolved by a multi-objective optimisation algorithm, where s_k represents a solution in the objective space. There is also an approximate true Pareto front set $tp = \{\ t_1, t_{2,} \ldots\ldots, t_{|tp|}\}$ in the objective space. Therefore, Inverted Generational Distance (IGD) is calculated by Eq. 4.6 for A using tp as the reference points (Ishibuchi et al. 2014). In Eq.4.6, $dist(t_i, s_k)$ is the Euclidean distance between t_i and s_k in the objective space. IGD in Eq.4.6 is the average distance from each point t_i in the approximate true Pareto front to its nearest solution in the approximation set A.

$$IGD(tp, A) = \frac{1}{|tp|} \sum_{i=1}^{tp} \min \{dis(\ t_i, s_k) | s_k \in A\}, \qquad (4.6)$$

Hypervolume (HV). HV measures the size of the objective space that the membered solutions of an approximation set A cover (Zitzler and Thiele 1999, Riquelme, Von Lucken, and Baran 2015). For each solution $s \in A,$ a hypercubic space rec_s is constructed using the reference point $refp$ and the solutions a as the diagonal corners of the hypercube (Saraswat and Saini

2013). $refp$ is defined as the vector of the worst objective function values from all dominated and non-dominated solutions. Therefore, HV is calculated as defined in Eq.4.7 (Saraswat and Saini 2013).

$$Hypervolume\ (HV) = volume\left(\bigcup_{s=1}^{|A|} rec_s\right) \tag{4.7}$$

Metrics concerning the classification performance are based on the values from confusion matrix as defined in TABLE 4.4 and are computed based on Eq. 4.8 for binary classification problems and Eq. 4.9 for multi-class classification problems.

TABLE 4.4

CONFUSION MATRIX

Actual Label	**Predicted Positive**	**Predicted Negative**
Actual Positive	True Positive (TP)	False Negative (FN)
Actual Negative	False Positive (FP)	True Negative (TN)

In the latter case, the macro-averaging values of these metrics are used (Asch 2013).

Sensitivity. Sensitivity measures the effectiveness of a classifier in identifying the positive labels, i.e. 1 in this project for a binary classification problem.

Specificity. Specificity measures the effectiveness of a classifier in identifying the negative labels, i.e. 0 in this project for a binary classification problem.

Accuracy. Accuracy measure the overall effectiveness of a classifier.

The number of misclassified data points. The number of misclassified data points is equal to the summation of all the data points that have been wrongly classified as positive and negative labels while their correct labels are negative and positive respectively.

As it can be seen in Eqs.4.8 and 4.9, True Positive (TP) represents the number of data points being correctly classified as label 1. True Negative (TN) represents the number of data points

that are correctly classified as label 0. False Negative (FN) represents the number of data points that are misclassified as label 0. False Positive (FP) represents the number of data points that are misclassified as label 1.

$$Sensitivity = \frac{TP}{TP + FN}$$

$$Specificity = \frac{TN}{TN + FP} \tag{4.8}$$

$$Accuracy = \frac{TN + TP}{TN + TP + FN + FP}$$

$$misclassified\_points = FN + FP$$

For multi-class classification problems, the average of the measures calculated for each binary classifies are used, which is known as macro-averaging. It is believed that in macro-averaging every classes will be treated equally instead of favouring the bigger classes (Asch 2013). In Eq.4.9, C, represents the total number of classes.

$$Sensitivity\_ma = \frac{\sum_{i=1}^{C} \frac{TP_i}{TP_i + FN_i}}{C}$$

$$Specificity\_ma = \frac{\sum_{i=1}^{C} \frac{TN_i}{TP_i + FP_i}}{C} \tag{4.9}$$

$$Accuracy\_ma = \frac{\sum_{i=1}^{C} \frac{TP_i + TN_i}{TP_i + FN_i + TN_i + FP_i}}{C}$$

$$misclassified\_points\_ma = \frac{\sum_{i=1}^{C} FN_i + FP_i}{C}$$

In the following sections, the implemented experiments will be discussed in more details and their corresponding computational results will be reported.

4.4.3 Selection of the Objective Functions

In this section, three different combinations of objective functions are investigated: 1) 'accuracy vs. interpretability', 2) 'accuracy vs. diversity (NCL)' and 3) 'accuracy vs. diversity (PFC)'. The most suitable combination that can result in a set of FRBSs of higher accuracy, more diversity and better interpretability will be chosen in this project.

To this purpose, IMOFM has been tested on the two benchmark datasets: the Australian credit card assessments dataset and the diabetes dataset. To provide rigorous analysis and comparison between different combinations of objective functions, each experiment was run 20 times. In each run, the training and testing sets based on the benchmark datasets are randomly partitioned according to the proportion mentioned in Section 4.4. For each experiment, the number of generations was set to 900 according to the parameter analysis reported in Section 4.4.4.1. The maximum and minimum number of rules are set to 10 and 2 respectively according to the parameter analysis reported in Section 4.4.4.2. Other parameters specific to the multi-objective immune algorithm are kept the same as those in (J. Chen and Mahfouf 2012). The details of the experiments for this section are summarised in Table 4.5.

TABLE 4.5

COMBINTAION OF OBJECTIVES: LIST OF EXPERIMENTS

Combination of Objectives	Benchmark Datasets	Numb of Runs
accuracy vs. interpretability	Australian Credit Card dataset	20
accuracy vs. interpretability	Diabetes dataset	20
accuracy vs. diversity (NCL)	Australian Credit Card dataset	20
accuracy vs. diversity (NCL)	Diabetes dataset	20
accuracy vs. diversity (PFC)	Australian Credit Card dataset	20
accuracy vs. diversity (PFC)	Diabetes dataset	20

Tables 4.6 and 4.7 present the results of IMOFM with three different combinations of objective functions using the Australian credit card assessment and diabetes datasets respectively.

TABLE 4.6

COMPARISIONS OF DIFFERENT COMBINATIONS OF OBJECTIVE FUNCTIONS ON THE AUSTRALIAN CREDIT CARD DATASET

Metrics	accuracy vs. interpretability	accuracy vs. NCL	accuracy vs. PFC
accuracy-best-tr (std.)	0.9177 (0.0149)	0.9155 (0.0164)	0.9243 (0.0144)
accuracy-best-ts (std.)	0.8448 (0.0285)	0.8512 (0.0300)	0.8477 (0.0296)
complexity-all (std.)	126.2339 (0.0590)	178.9169 (0.1187)	241.5541 (0.1657)
diversity-all (std.)	0.2183 (0.0477)	0.5123 (0.0521)	0.4651 (0.0854)
accuracy-all-tr (std.)	0.9007 (0.0251)	0.7168 (0.1116)	0.8078 (0.0704)
accuracy-all-ts (std.)	0.8476 (0.0229)	0.6837 (0.1090)	0.7531 (0.0685)

The obtained FRBSs from each combination of objective functions were averaged over 20 runs. In each run, the obtained FRBSs on the approximate Pareto front using the training (tr) and testing (ts) datasets are analysed based on their classification accuracy (see Eq. 4.8, complexity (see Eq.4.3) and diversity (see Eq. 4.5).

TABLE 4.7

COMPARISIONS OF DIFFERENT COMBINATION OF OBJECTIVE FUNCTIONS ON THE DIABETES SET

Metrics	accuracy vs. interpretability	accuracy vs. NCL	accuracy vs. PFC
accuracy-best-tr (std.)	0.8245 (0.0153)	0.8157 (0.0138)	0.8204 (0.0161)
accuracy-best-ts (std.)	0.7565 (0.0309)	0.7583 (0.0223)	0.7604 (0.0281)
Complexity (std.)	58.7086 (0.0372)	78.1979 (0.0604)	105.9388 (0.0653)
diversity-PFC (std.)	0.1969 (0.0294)	0.6014 (0.0153)	0.5097 (0.0449)
accuracy-all-tr (std.)	0.8030 (0.0096)	0.6457 (0.0382)	0.7070 (0.0306)
accuracy-all-ts (std.)	0.7649 (0.0078)	0.6286 (0.0391)	0.6711(0.0257)

In order to compute accuracy-best, in each run, among all the FRBSs on the Pareto front, the FRBS with the least RMSE will be selected. The classification accuracy of the selected best models will then be computed. Finally, the classification accuracy of the best model in each run will be summed up and averaged across the 20 runs. In order to compute complexity-all, in each run, complexity of all the Pareto front solutions is individually computed. All the individual complexities are then summed and averaged over all the Pareto solutions in that run,

which will then be averaged across all 20 runs. diversity-all (PFC) computes PFC between the FRBSs in the Pareto front at each run, which will then be averaged across all 20 runs. For all the experiments carried out in this work, PFC is chosen as the measure of diversity since PFC has shown to be a more effective measure of diversity comparing to NCL (Bhowan et al. 2013). accuracy-all computes the averaged classification accuracy for all the FRBSs in the Pareto front at each run, which will then be averaged over 20 runs. For all the metrics reported in the tables, their respective standard deviations (std.) are also included.

As mentioned in Section 4.2.3, the ultimate aim of employing a multi-objective fuzzy classification approach, such as IMOFM, is to design a well-performed ensemble classifier. Therefore, generating diverse yet accurate enough classifiers is the aim of this chapter and lays firm foundation for Chapter 5. In light of this aim, in this chapter, we will choose the combination of objective functions which can lead to a set of diverse, accurate and interpretable FRBSs.

From Tables 4.6 and 4.7, it can be seen that using either 'accuracy vs. diversity (NCL)' or 'accuracy vs. diversity (PFC) can result in higher accuracy for the best model on the testing set. However, there are two main factors which require our attention before drawing any further conclusions on choosing these two combinations as the objective functions. First, employing these two combinations of objective functions led to a higher number of Pareto FRBSs in each run. More than half of the obtained FRBSs performed poorly in terms of their RMSE, which makes them highly inaccurate for the classification purpose. Second, the computational time associated with these two combinations is relatively higher than that of 'accuracy vs. interpretability'. In light of the above two reasons, these two combinations of objective functions, i.e. 'accuracy vs. diversity (NCL)', and 'accuracy vs. diversity (PFC), are not preferable to be utilised as the objective functions in designing an ensemble of classifier.

The next observation from the results is regarding the diversity values of the evolved FRBSs. The emphasise of ‘accuracy vs. diversity (NCL)’ and ‘accuracy vs. diversity (PFC)’ is more on the diversifying of the solutions rather than encouraging the accuracy. Therefore, the evolved FRBSs can be significantly different from each other. However, as neither NCL nor PFC are not directly linked to the permutation operators, i.e. rule and fuzzy set reduction operators, in IMOFM, the accuracy-all performance of the resulting FRBSs is sacrificed in exchange for a better diversity performance. However, it is worth mentioning that PFC or NCL may serve as a good diversity preserving mechanism in a metaheuristic based multi-objective optimisation framework, which will be discussed further in Chapter 5.

It can be seen from both tables that using ‘accuracy vs. interpretability’ as the objective functions can result in higher accuracy-all in the obtained FRBSs, meaning that more FRBSs on the Pareto front can be used at the ensemble classification stage that will further improve the overall classification accuracy. It can also be seen from the results that ‘accuracy vs. interpretability’ not only encouraged the diversity of the solutions to some extent and decreased the complexity of the FRBSs but also evolved significantly more FRBSs with higher accuracy comparing to the other two combination of objectives.

In light of the above discussions, ‘accuracy vs. interpretability’ is chosen as the objective functions in all the experiments of this work.

4.4.4 Parameter Analysis

In this section, the sufficient number of generations to warrant a convergence of the IMOFM based classification mechanism and the predefined maximum and minimum number of rules in IMOFM are investigated. For the parameter analysis experiments in the following sections,

the benchmark datasets, i.e. the Australian credit card assessment dataset and diabetes dataset, are employed.

4.4.4.1 The Number of Generations

The number of generations is served as the stopping criteria when there is no significant improvement in the two performance measures, e.g. Hypervolume (HV) and Inverted Generational Distance (IGD) measures.

For the implementation of the experiments in this section, IMOFM has been tested on the two benchmark datasets. To provide statistical analysis, all experiments were run 20 times respectively. For each run, the same training and testing sets (see Section 4.4.1) were employed. For all experiments, 'accuracy vs. interpretability' is formulated as the objective functions. The number of generations was set to 1200 initially for all experiments. The maximum and minimum number of rules were set to 10 and 2 respectively according to the experiment results provided in Section 4.4.4.2. Other parameters specific to the multi-objective immune algorithm are kept the same as those in (J. Chen and Mahfouf 2012).

TABLE 4.8

HYPERVOLUME VALUES-AUSTRALIAN CREDIT CARD

Gen	Mean-HV	STD-HV	Max-HV	Min-HV	HV-nondom	HV-true
100	92.3986	0.2318	99.0732	87.2879	101.8467	119.6805
200	94.2979	0.2287	102.4948	89.5544	104.0184	119.68.05
300	95.9963	0.2238	105.1313	91.0895	105.5529	119.6805
400	97.7909	0.2255	107.2104	91.5517	109.8897	119.6805
500	99.2246	0.2188	110.3287	93.2372	112.0593	119.6805
600	100.1130	0.2213	110.7504	93.4918	113.7796	119.6805
700	100.8228	0.2311	111.3443	93.7965	117.1403	119.6805
800	101.3768	0.2357	111.7215	94.7745	117.3050	119.6805
900	101.8837	0.2368	112.8497	94.9155	118.3837	119.6805
1000	102.1841	0.2354	113.8931	95.2499	118.4098	119.6805
1100	102.4488	0.2328	114.7504	95.2361	119.5147	119.6805
1200	102.6067	0.2334	114.9549	95.4948	119.6805	119.6805

The results of this experiment were analysed over 20 runs from generation 100 to generation 1200.

Tables 4.8-4.11 present the HV and IGD values respectively for the Australian credit card assessment dataset and the diabetes dataset.

TABLE 4.9

INVERTED-GENERATIONAL-DISTANCE VALUES-AUSTRALIAN CREDIT CARD

Gen	Mean-IGD	STD-IGD	Max-IGD	Min-IGD	IGD-nondom	IGD-true
100	3.2581	0.2518	4.0630	2.7487	2.6379	0
200	3.2573	0.2101	7.0837	2.3896	2.2744	0
300	3.3237	0.2668	7.3028	1.9322	1.9238	0
400	3.2836	0.2900	7.3466	1.3256	1.1581	0
500	3.1413	0.2887	7.3264	0.9657	0.8571	0
600	3.1850	0.3028	7.3004	0.9273	0.5890	0
700	3.1402	0.2927	7.6384	0.8356	0.3603	0
800	3.1460	0.2925	7.5988	0.6988	0.2321	0
900	3.1287	0.2955	7.5965	0.6435	0.2000	0
1000	3.1191	0.2967	7.4332	0.5701	0.1565	0
1100	3.1666	0.3047	7.5978	0.8385	0.1212	0
1200	3.3028	0.3195	7.4932	0.9233	0	0

TABLE 4.10

HYPERVOLUME VALUES-DIABETES

Gen	Mean-HV	STD-HV	Max-HV	Min-HV	HV-nondom	HV-True
100	14.8724	0.3131	16.2215	13.5863	16.6835	19.8575
200	15.4792	0.2958	16.7643	13.6695	17.2817	19.8575
300	15.9055	0.2873	17.3481	13.7951	17.9578	19.8575
400	16.1983	0.2582	17.8947	13.8545	18.3729	19.8575
500	16.4493	0.2606	17.9879	13,8453	18.6108	19.8575
600	16.6024	0.2592	18.0892	13.9067	18.7658	19.8575
700	16.7490	0.2561	18.2409	13.9106	19.1011	19.8575
800	16.9046	0.2575	18.3212	13.9634	19.2434	19.8575
900	17.0516	0.2616	18.3802	13.9572	19.5666	19.8575
1000	17.1389	0.2632	18.4140	13.9456	19.6733	19.8575
1100	17.2218	0.2571	18.5258	13.9393	19.7597	19.8575
1200	17.2986	0.2560	18.5899	13.9302	19.8575	19.8575

TABLE 4.11

INVERTED-GENERATIONAL-DISTANCE VALUES-DIABETES

Gen	Mean-IGD	STD-IGD	Max-IGD	Min-IGD	IGD-nondom	IGD-True
100	2.4105	0.3011	3.0909	0.4279	1.6719	0
200	2.4867	0.2478	4.2651	1.8503	1.3203	0
300	2.5370	0.2568	4.2537	1.6547	1.2843	0
400	2.5910	0.2936	4.2457	1.6319	1.1110	0
500	2.5656	0.2845	4.2547	1.4607	0.9110	0
600	2.5969	0.2981	4.2616	1.4392	0.4895	0
700	2.5627	0.2967	4.2381	1.4075	0.4507	0
800	2.5984	0.2775	4.4975	1.4216	0.3062	0
900	2.6106	0.2780	4.4756	1.4361	0.2602	0
1000	2.6094	0.2743	4.4756	1.2381	0.2602	0
1100	2.6360	0.2863	4.5628	1.2700	0.0932	0
1200	2.6353	0.2868	4.5628	1.2700	0	0

In Tables 4.8 and 4.10, the average of HV is computed for each generation under consideration across 20 runs using Eq. 4.6 for all the evolved FRBSs at the Pareto front. STD-HV, Max-HV and Min-HV, represent the standard deviation, maximum and minimum values of the computed HVs.

To compute HV, the reference points are defined as a vector of the worst objective function values from all dominated and non-dominated solution from generation 100 to generation 1200 across all the 20 runs.

In Tables 4.8 and 4.10, for calculation of HV-nondom values for each generation, all the FRBSs at the Pareto front from each run will be put together and then the nondominated FRBSs will be identified representing the approximate Pareto front for that particular generation (e.g. Gen 100). HV-true is the HV value computed for the approximate true Pareto front. To compute the approximate true Pareto front for each dataset, all the FRBSs in the approximate Pareto front from generation 100 to generation 1200 will be put together and then the identified nondominated FRBSs will represent the approximate true Pareto front.

Tables 4.9 and 4.11 present the IGD values for the Australian credit card and diabetes datasets respectively. The average of IGD across 20 runs is computed for each generation under consideration using Eq. 4.7 for all the evolved FRBSs at the Pareto front, which is represented as Mean-IGD in these tables. STD-IGD, Max-IGD and Min-IGD represent the standard deviation, maximum and minimum values of the computed IGDs respectively. IGD-nondom for each generation, is calculated for the FRBSs in the approximate Pareto front for that particular generation. In computing IGD for both benchmark datasets, the approximate true Pareto fronts are the ones explained earlier in computation of HV.

As it can be seen from the results, generation 900 for both benchmark datasets, has proved to be the best trade-off between the computational expense and the quality of the solutions. After 900, there are no significant improvements in either metrics for both datasets.

In light of the above discussions, 900 is chosen in all the remaining experiments of this thesis.

In the next Section, the maximum and minimum number of rules for IMOFM is investigated, and the computational results are provided.

4.4.4.2 The Maximum and Minimum Number of Rules

The aim of this section is to identify the effect of this number on the performance of IMOFM classification mechanism.

In this section, experiments were implemented using two benchmark datasets. We keep the optimal parameter settings the same as the results from previous sections. Other parameters specific to the multi-objective immune algorithm are kept the same as those in (J. Chen and Mahfouf 2012).

The results of 10 different combinations (see Table 4.12) of the maximum and minimum number of rules for the Australian credit card assessments dataset and the diabetes dataset are presented in Tables 4.13-4.16. The performance metrics including HV and IGD are computed using Eqs. 4.6 and 4.7 in a similar way as in Section 4.4.2 Mean-HV and Mean-IGD are both computed the same way as in Section 4.4.2 To compute HV-nondom and IGD-nondom, for each combination, the approximate Pareto front for that combination (e.g. Max 10-Min 2) will be identified and the corresponding HV and IGD values are denoted as HV-nondom and IGD-nondom (see Tables 4.13 and 4.15). The approximate true Pareto front for each dataset is the nondominated front of the approximate Pareto fronts from all the combinations, from which HV-true will be computed.

To further analyse the results obtained from this experiment, accuracy is computed using Eq. 4.9 for both training and testing sets for all the combinations (see Tables 4.14 and 4.16). For each combination, Mean-Best-ACC-Training and Mean-Best-ACC-Testing are computed by the following three steps: 1) at each run, among all the FRBSs at the Pareto front, the FRBS with the least RMSE is selected; 2) the classification accuracy of the selected FRBS will then be computed; and 3) the computed accuracy will be averaged across 20 runs.

TABLE 4.12

THE MAXIMUM AND MINIMUM NUMBER OF RULES: LIST OF EXPERIMENTS

Max-Min Rules	**Datasets**	**Numb of Runs**
Max 8-Min 2	Australian Credit Card dataset & Diabetes dataset	20
Max 8-Min 4	Australian Credit Card dataset & Diabetes dataset	20
Max 10-Min 2	Australian Credit Card dataset & Diabetes dataset	20
Max 10-Min 5	Australian Credit Card dataset & Diabetes dataset	20
Max 12-Min 2	Australian Credit Card dataset & Diabetes dataset	20
Max 12-Min 6	Australian Credit Card dataset & Diabetes dataset	20
Max 14-Min 2	Australian Credit Card dataset & Diabetes dataset	20
Max 14-Min 7	Australian Credit Card dataset & Diabetes dataset	20
Max 16-Min 2	Australian Credit Card dataset & Diabetes dataset	20
Max 16-Min 8	Australian Credit Card dataset & Diabetes dataset	20

TABLE 4.13

HYPERVOLUME VALUES-AUSTRALIAN CREDIT CARD

Max-Min Rules	mean-HV	HV-nondom	HV-true	mean-IGD	IGD-nondom	IGD-true
Max 8-Min 2	181.3145	198.2026	215.7270	3.5990	2.7327	0
Max 8-Min 4	160.8338	185.8350	215.7270	9.3560	7.1197	0
Max 10-Min 2	177.6033	199.5871	215.7270	3.9595	3.0547	0
Max 10-Min 5	155.3073	172.0046	215.7270	11.8862	10.6370	0
Max 12-Min 2	186.0284	205.3401	215.7270	2.1800	2.6492	0
Max 12-Min 6	146.2083	159.5339	215.7270	15.1215	13.4917	0
Max 14-Min 2	183.7657	202.9868	215.7270	2.8549	1.2662	0
Max 14-Min 7	136.9855	153.4587	215.7270	18.5077	16.4105	0
Max 16-Min 2	181.0341	200.2389	215.7270	3.3579	2.0166	0
Max 16-Min 8	130.4159	144.9955	215.7270	21.5353	17.7254	0

TABLE 4.14

MEAN-BEST-ACCURACY FOR TRAINING & TESING- AUSTRALIAN CREDIT CAR

Max-Min Rules	Mean-Best-ACC-Training	Mean-Best-ACC-Testing
Max 8-Min 2	0.9241	0.8471
Max 8-Min 4	0.9251	0.8448
Max 10-Min 2	0.9171	0.8448
Max 10-Min 5	0.9332	0.8448
Max 12-Min 2	0.9222	0.8474
Max 12-Min 6	0.9347	0.8462
Max 14-Min 2	0.9203	0.8497
Max 14-Min 7	0.9366	0.8494
Max 16-Min 2	0.9317	0.8451
Max 16-Min 8	0.9417	0.8448

TABLE 4.15

HYPERVOLUME AND INVERTED GENERATIONAL DISTANCE VAUES-DIABETES

Max-Min Rules	mean-HV	HV-nondom	HV-true	mean-IGD	IGD-nondom	IGD-true
Max 8-Min 2	43.9750	47.9922	54.1647	4.4272	5.1278	0
Max 8-Min 4	38.6719	43.2707	54.1647	4.8582	2.6811	0
Max 10-Min 2	43.9342	48.5179	54.1647	3.7012	1.6696	0
Max 10-Min 5	36.4606	40.0594	54.1647	4.9675	3.7308	0
Max 12-Min 2	46.0185	49.9270	54.1647	3.3351	1.0233	0
Max 12-Min 6	34.3757	39.1465	54.1647	5.6419	4.1815	0
Max 14-Min 2	46.2516	50.4058	54.1647	2.4880	1.0100	0
Max 14-Min 7	31.8621	36.0557	54.1647	6.6938	5.8132	0
Max 16-Min 2	43.9962	49.9938	54.1647	2.5075	0.5909	0
Max 16-Min 8	29.5458	34.9960	54.1647	7.9792	6.5033	0

TABLE 4.16
MEAN-BEST-ACCURACY FOR TRAINING & TESING-DIABETES

Max-Min Rules	Mean-Best-ACC-Training	Mean-Best-ACC-Testing
Max 8-Min 2	0.8213	0.7591
Max 8-Min 4	0.8271	0.7542
Max 10-Min 2	0.8245	0.7565
Max 10-Min 5	0.8345	0.7581
Max 12-Min 2	0.8172	0.7594
Max 12-Min 6	0.8394	0.7531
Max 14-Min 2	0.8215	0.7523
Max 14-Min 7	0.8457	0.7507
Max 16-Min 2	0.8285	0.7583
Max 16-Min 8	0.8466	0.7458

The results in Tables 4.13-4.16 confirm that when the minimum number of rules is 2, the obtained Pareto fronts cover a wider space of the approximate true Pareto front. On the other hand, when the minimum number of rules is half of the maximum number of rules, the obtained Pareto fronts cover only a smaller fraction of the approximate true Pareto front. Therefore, having the minimum number of rules set to 2 will encourage the search algorithm to explore a wider space and find more diverse solutions which may be useful for ensemble modelling. It is worth mentioning that if the search space is reduced, e.g. by setting the minimum number of rules to half of the maximum number of rules, the chance of finding a solution with a higher accuracy is increased. This factor indeed inspires the incorporation of the preference-based search concept in Chapter 5 to focus the search on the region of the Pareto front that can achieve both diversity and accuracy.

In Figure 4.2, the Pareto fronts of a randomly chosen run for two of the scenarios, i.e. Max 10-Min 2 and Max 10-Min 5, are plotted against the approximate true Pareto fronts for both benchmark datasets. The minimum number of rules is set to 2 in this work for the following reasons that can be clearly seen from Tables 4.13-4.16 and Figure 4.1.

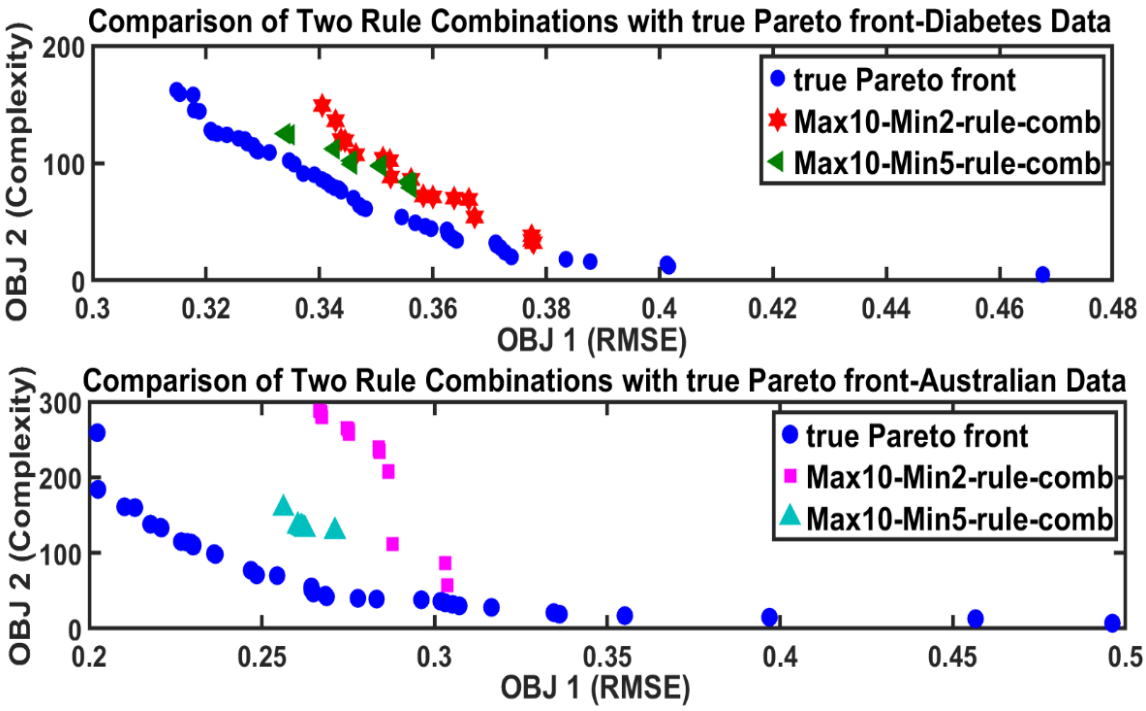

Figure 4.2 Comparison of the Pareto fronts obtained from a randomly chosen run.

(1) Keeping the minimum number of rules to 2 encourages the diversity among all the solutions as a wider search space can be explored.(2) Although the obtained Pareto fronts may end up closer to the approximated true Pareto front when the minimum number of the rules is set to the half of the maximum number of the rules, the scenarios of e.g. Max 10-Min 2 can still achieve comparable performance measured by RMSE.

In order to compare and visualise the impact of the choice of the maximum number of rules on the convergence of the proposed classification mechanism, the minimum number of rules is set to 2 and the performance metrics including Mean-HV, HV-nondom, Mean-IGD and IGD-nondom are illustrated as Boxplots in Figures 4.3 and 4.4 for the Australian credit card assessments and diabetes datasets respectively.

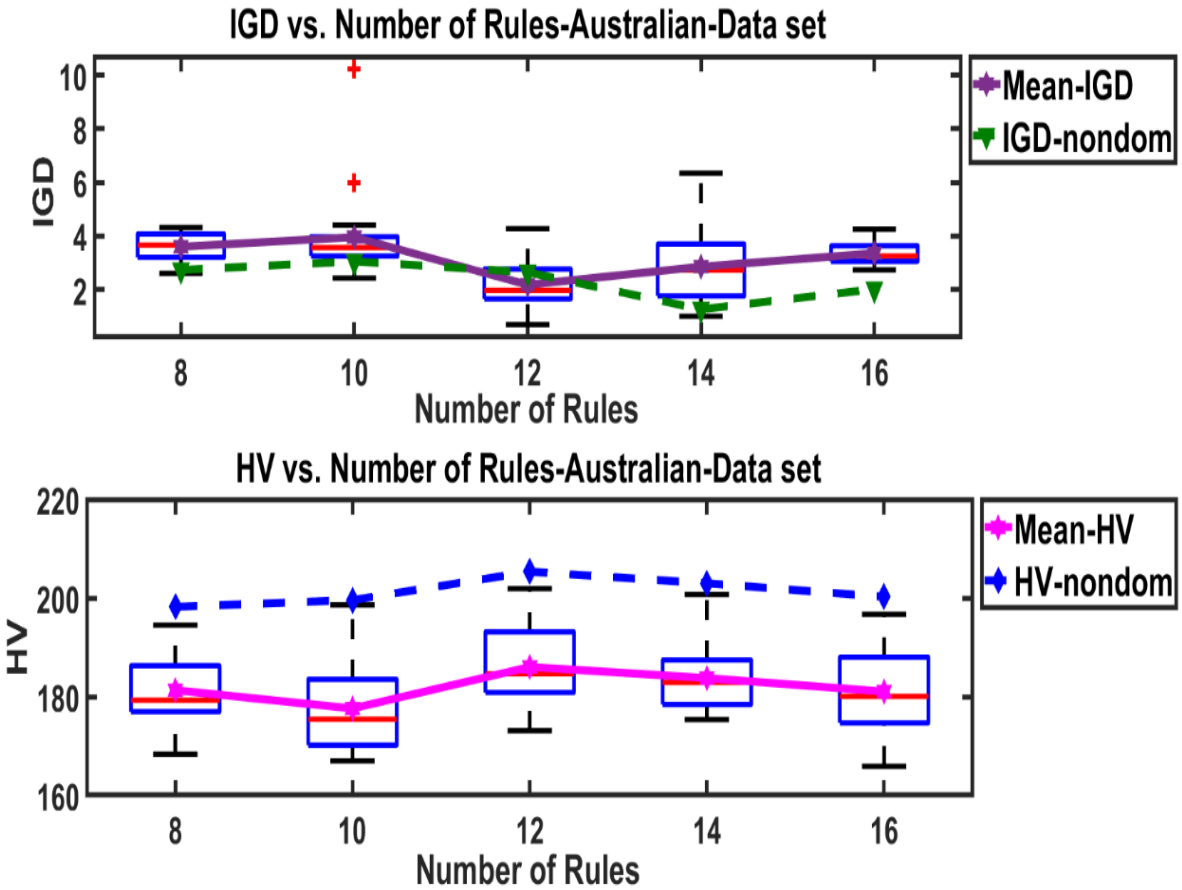

Figure 4.3 Boxplots of performance metrics including HV and IGD for the Australian credit card assessment dataset based on the values reported in Table 4.13 against the maximum number of rules.

Figure 4.3 seemly suggests that the maximum number of rules should be set to 12 as this setting leads to lower Mean-IGD and IGD-nondom values and higher Mean-HV and HV-nondom values comparing to the other settings. However, the performance difference between 10 rules and 12 rules is not significant.

This observation is further confirmed by Figure 4.4, where setting the maximum number of rules to 10 shows a start of improvement compared to 8 rules in Mean-IGD and IGD-nondom metrics. However, this trend becomes steady with no significant improvement by setting the maximum number of rules to a higher value.

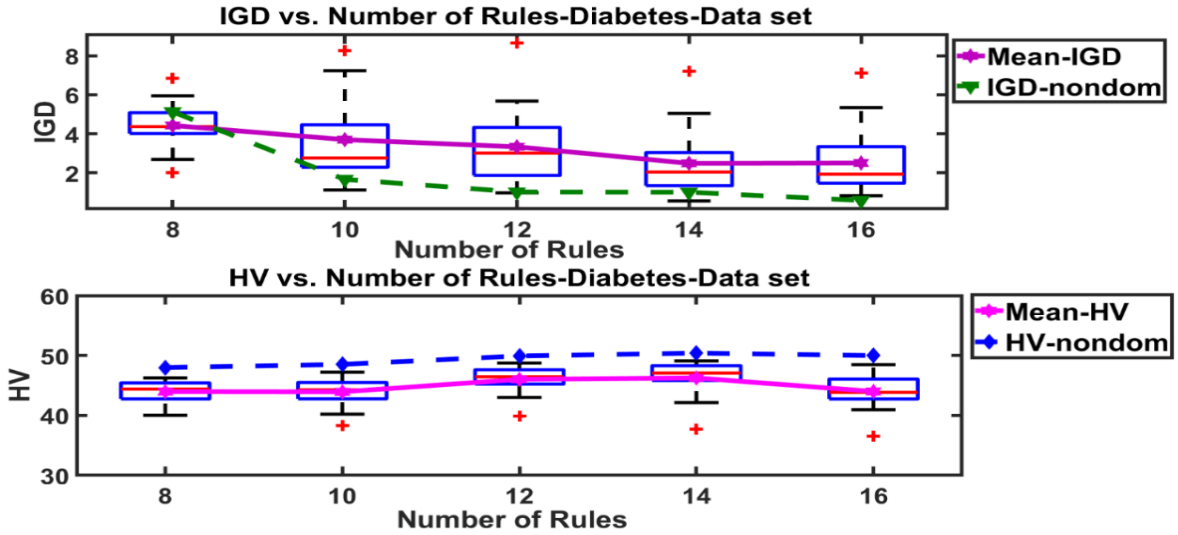

Figure 4.4 Boxplots of performance metrics including HV and IGD for the diabetes dataset based on the values reported in Table 4.15 against the maximum number of rules.

In order to compare and visualise the impact of the choice of the maximum number of rules on the classification accuracy of the proposed classification mechanism, the minimum number of rules is set to 2 and the classification accuracy are illustrated as Boxplots in Figures 4.5 and 4.6 for the Australian credit card assessments and diabetes datasets respectively.

Figure 4.5 Boxplots of classification accuracy for the training and testing sets of the Australian credit card assessment dataset based on the values in Table 4.14 against the maximum number of rules.

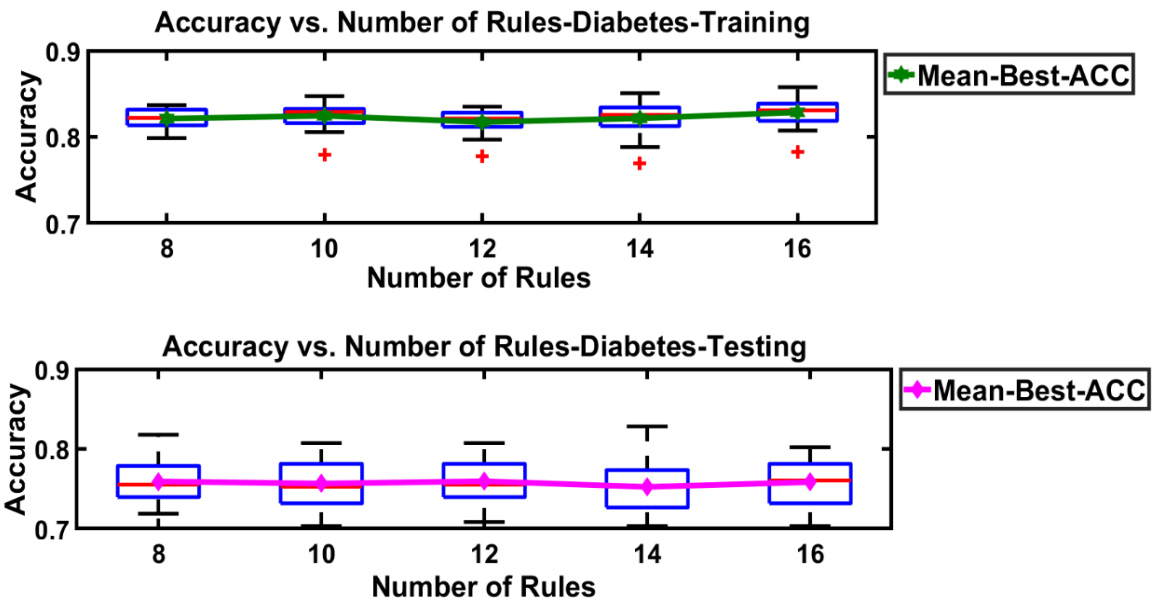

Figure 4.6 Boxplots of classification accuracy for the training and testing sets of the diabetes data based on the values reported in Table 4.16 against the maximum number of rules.

Also, in Figures 4.5 and 4.6 show a rather steady performance in terms of classification accuracy across all five settings for the maximum number of rules. It is worth pointing out that for the diabetes training sets, setting the maximum number of rules to 10 leads to the second highest performance in terms of Mean-Best-ACC-training although the differences from those of the other settings are not significant.

In light of the above observations and taking into account the computational time of executing IMOFM, the maximum number of rules is set 10 in this work. Finally, the setting of 'Max 10-Min 2' is chosen in this work based on two main reasons. 1) It is computationally more efficient comparing to the settings of the higher maximum number of rules. 2) 'Max 10-Min 2' allows the IMOFM algorithm to explore more search space and therefore increases the diversity of the solutions comparing to 'Max 8- Min 2'. In the next section, the performance of the IMOFM classification mechanism using the above identified objective functions and parameters is

compared with the baseline classification algorithms including ANNs, ANFIS and CART on binary classification problems.

4.4.3 Results on Binary Classification

For the implementation of the experiments in this section, the recorded EEG datasets from 4 subjects were employed and the properties of the datasets for each subject and their corresponding classifiers are summarised in Table 4.3. For each subject, six OAO-based binary classifiers (i.e.CF1, CF2, CF3, CF4, CF5, CF6) were trained.

The two benchmark datasets, i.e. the Australian credit card assessment, and Diabetes dataset are also used in this section to test the performances of the developed the IMOFM classification mechanism. The properties of these datasets are summarised in Table 4.2. For the Australian credit card dataset, Class 1 represents the credit awarded and Class 2 represents otherwise. In the diabetes dataset, Class 1 corresponds to a patient diagnosed with diabetes and Class 2 corresponds to a patient diagnosed otherwise. For each benchmark datasets, a OAO-binary classifiers were trained.

To provide a rigorous analysis and comparison with other approaches for the benchmark datasets and EEG datasets for each subject, training of each classifier was run 10 times. For each run, the training and testing sets are randomly partitioned into 75% training and 25% testing.

For comparison purpose, baseline algorithms including ANNs, ANFIS, CART, and SVM implemented in MATLAB® toolboxes were also run 10 times using the same training and testing sets. The parameters used for the ANNs algorithm were kept the same as the default setting. However, in order to make a fair comparison with the IMOFM-C algorithm, preliminary experiments were carried out to choose an optimal number of hidden layers and

neurons by trial and error. Therefore, the number of hidden layers was set to 1 with 20 neurons and the number of epochs was set to 900. For ANFIS, sub-clustering was employed to generate the initial fuzzy structure with 10 rules. In order to make sure, the generated fuzzy structure has the same number of rules as in IMOFM-C (i.e. 10 rules), the cluster influence range were manually tuned in the range of [0.29, 0.5]. The hybrid method was selected as the optimisation method and the number of epochs was set to 900. All other parameters were kept the same as the default setting of the ANFIS toolbox. For implementation of the CART decision tree, based on the empirical studies, Fine tree was used as the classifier type, which uses as many leaves as possible to make fine distinctions between the classes. The maximum number of splits for the Fine tree classifier is 100. For implementation of SVM, Fine Gaussian SVM is used.

Since the IMOFM classification mechanism is based on the multi-objective optimisation fuzzy modelling approach, the implementation of this classification algorithm, results in evolving a Pareto front of solutions at each run. In order to select a FRBS as the final classifier from the set of Pareto FRBSs and compare with baseline classification approaches, an evaluation procedure has been employed as follows. 1) In each run, from the nondominated FRBSs, the one with the lowest RMSE value was selected as the best model. 2) The selected FRBS together with the post-processing procedure mentioned in Section 4.2.1 was used as the classifier and was evaluated against the training and testing sets using different classification performance measures, including Sensitivity, Specificity Accuracy and the total number of miss-classified data points as computed in Eq. 4.9

For the both benchmark datasets, the classification performance for each algorithm on the training and testing sets was evaluated by the four classification performance metrics. For the EEG datasets, the classification performance for each algorithm on the training and testing sets for each subject was evaluated by the four classification performance metrics over six

classifiers. In the benchmark and the EEG datasets the averaged results across 10 runs are reported here.

The comparison of performance of IMOFM classification mechanism, with the baseline algorithms on the training (TR) and testing (TS) sets for the benchmark datasets and EEG datasets from 4 subjects are presented in Tables 4.17- 4.22, respectively.

The IMOFM classification mechanism performs consistently better in the four classification performance metrics on the testing sets for the benchmark and the EEG datasets comparing to the four baseline classification algorithms. Removing redundant rules and membership functions is responsible for the improved generalisation capability on the testing sets. It is worth pointing out that all baseline algorithms tend to overfit the models to the training datasets, leading to the worse performance on the testing datasets.

TABLE 4.17

COMPARSION OF CLASSIFICATION PERFORMANCE OF TRAINING AND TESTING ON AUSTRALIAN DATASET

Classifiers	**Sensitivity (TR/TS)**	**Specificity (TR/TS)**	**Accuracy (TR/TS)**	**Miss-points (TR/TS)**
IMOFM-C	0.9221/**0.8462**	0.9221/**0.8462**	0.9245/**0.8477**	39.10/**26.20**
ANNs	1/0.8119	1/0.8119	1/0.8116	0/32.40
ANFIS	0.9518/0.8230	0.9518/0.8230	0.9525/0.8238	24.60/30.30
CART	0.9470/0.8345	0.9470/0.8345	0.9483/0.8372	26.80/28
SVM	0.9889/0.6182	0.9889/0.6182	0.9896/0.6355	5.40/62.70

TABLE 4.18

COMPARSION OF CLASSIFICATION PERFORMANCE OF TRAINING AND TESTING ON DIABETS DATASET

Classifiers	**Sensitivity (TR/TS)**	**Specificity (TR/TS)**	**Accuracy (TR/TS)**	**Miss-points (TR/TS)**
IMOFM	0.7955/**0.7118**	0.7955/**0.7118**	0.8267/**0.7469**	99.80/**48.60**
ANNs	0.9942/0.6519	0.9942/0.6519	0.9951/0.6750	2.8/62.40
ANFIS	0.8227/0.7002	0.8227/0.7002	0.8516/0.7354	85.50/50.80
CART	0.9175/0.6733	0.9175/0.6733	0.9286/0.7005	41.100/57.50
SVM	0.9872/0.5185	0.9872/0.5185	0.9911/0.6411	5.100/68.90

TABLE 4.19

COMPARSION OF CLASSIFICATION PERFORMANCE OF TRAINING AND TESTING SUBJECT 1

Classifiers	Sensitivity (TR/TS)	Specificity (TR/TS)	Accuracy (TR/TS)	Miss-points (TR/TS)
IMOFM	0.8532/**0.6878**	0.8532/**0.6878**	0.8636/**0.7138**	20.8167/**15.30**
ANNs	1/0.6552	1/0.6552	1/0.6779	0/17.10
ANFIS	0.9844/0.6025	0.9844/0.6025	0.9848/0.6245	2.70/29.83
CART	0.9077/0.6340	0.9077/0.6340	0.9181/0.6640	13.25/17.9167
SVM	0.9605/0.5748	0.9605/0.5748	0.9679/0.6415	5.16/19.08

TABLE 4.20

COMPARSION OF CLASSIFICATION PERFORMANCE OF TRAINING AND TESTING SUBJECT 2

Classifiers	Sensitivity (TR/TS)	Specificity (TR/TS)	Accuracy (TR/TS)	Miss-points (TR/TS)
IMOFM	0.9336/**0.8660**	0.9336/**0.8660**	0.9362/**0.8748**	11.26/**10.83**
ANNs	1/0.8403	1/0.8403	1/0.8490	0/11
ANFIS	0.9834/0.8125	0.9834/0.8125	0.9834/0.8257	3.83/12.51
CART	0.9526/0.8278	0.9526/0.8287	0.9555/0.8417	7.98/11.23
SVM	0.9768/0.7640	0.9768/0.7640	0.9789/0.8131	4.70/12.80

TABLE 4.21

COMPARSION OF CLASSIFICATION PERFORMANCE OF TRAINING AND TESTING SUBJECT 3

Classifiers	Sensitivity (TR/TS)	Specificity (TR/TS)	Accuracy (TR/TS)	Miss-points (TR/TS)
IMOFM	0.8469/**0.7882**	0.8469/**07882**	0.8594/**0.8105**	58.88/**25.81**
ANNs	0.9591/0.7451	0.9591/0.7451	0.9593/0.7700	17.7/30.95
ANFIS	0.8990/0.7515	0.8990/0.7515	0.9047/0.7765	40.5500/30.08
CART	0.9360/0.7515	0.9360/0.7515	0.9440/0.7769	22.55/30.38
SVM	0.9384/0.7316	0.9384/0.7316	0.9518/0.7863	19.40/28.86

TABLE 4.22

COMPARSION OF CLASSIFICATION PERFORMANCE OF TRAINING AND TESTING SUBJECT 4

Classifiers	Sensitivity (TR/TS)	Specificity (TR/TS)	Accuracy (TR/TS)	Miss-points (TR/TS)
IMOFM	0.8671/**0.7351**	0.8671/**0.7351**	0.8686/**0.7405**	23.66/**15.30**
ANNs	0.9987/0.7167	0.9987/0.7167	0.9987/0.7250	0.25/16.08
ANFIS	0.9719/0.6876	0.9719/0.6876	0.9722/0.6925	5.18/17.83
CART	0.9237/0.7309	0.9237/0.7309	0.9280/0.7382	12.40 /15.51
SVM	0.9566/0.7153	0.9566/0.7153	0.9582/0.7364	7.26/15.56

Although we choose the best FRBS as the one with the minimum RMSE on the training set, this does not guarantee that it will lead to the best performance on the testing set.

In some cases, the classification accuracy of the best FRBS on the testing set is lower than the worst FRBS in the same Pareto front. An example of this observation is provided in Table 4.23 based on the CF1 classifier for Subject 1 for all 10 runs.

As it can be seen in TABLE 4.23, in Run 1, Run 3, Run 4, Run 7, and Run 8, the classification accuracy of the best FRBS on the testing set is lower than that of the worst FRBS. This observation indicates that in order to derive consistent classification results, there is a need to utilise multiple FRBSs on the Pareto front (as will be discussed in the next chapter) as we don't know which one is the best for the testing set.

In section 4.4.5 the performance of the IMOFM classification mechanism on the multi-class problem is compared to the baseline algorithms including ANNs, ANFIS, CART, and SVM.

TABLE 4.23

PERFORMANCE OF BEST AND WORST FRBSs FOR CF1-SUBJECT 1TRAINING AND TESTING

Runs	Best-ACC-TR	Best-ACC-TS	Worst-ACC-TR	Worst-ACC-TS
Run 1	0.8780	0.6341	0.8049	0.7317
Run 2	0.8943	0.7561	0.8049	0.7073
Run 3	0.8455	0.6829	0.8293	0.7073
Run 4	0.8699	0.6341	0.8130	0.7317
Run 5	0.8618	0.7805	0.8211	0.6829
Run 6	0.8699	0.7073	0.6423	0.6341
Run 7	0.9106	0.6098	0.7967	0.7805
Run 8	0.8293	0.7073	0.8130	0.8049
Run 9	0.9106	0.7073	0.8455	0.7073
Run 10	0.8130	0.6585	0.8618	0.6585

4.4.4 Interpretability of the IMOFM-C Algorithm on Binary Classification

As it mentioned earlier in this chapter, the interpretability of the IMOFM-C classification algorithm is one of the significant features of this approach. Therefore, Figure 4.7 illustrates

the rule-based of the simplest classifiers among the pareto of nondominated CF1 classifiers from one of the 10runs for Subject 3.

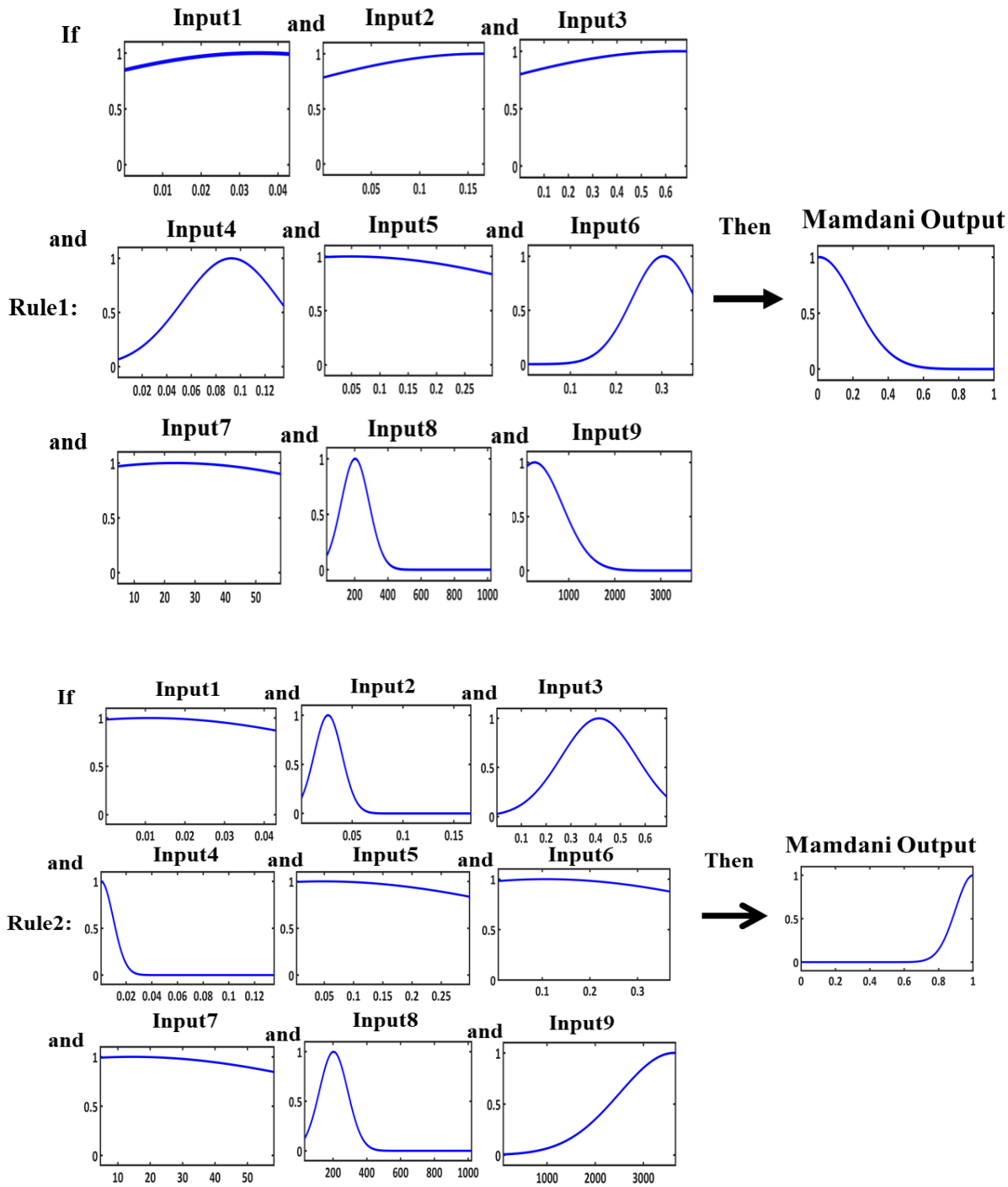

Figure 4.7 An example of the rule-base for the simplest classifier chosen from the nondominated CF1 classifiers of Subject 3 from one of the ten runs.

The chosen classifier has two rules. Input 1 to 9 represent the total of nine features. They are referred to as the relative energy, entropy and mean power of detailed coefficients, which are respectively computed for the three frequency bands, i.e. gamma, beta, and alpha. The consequent of each rule represents the certainty of prediction for a given data point across the range of [0,1]. The higher the firing strength associated with one of the rules, the higher the certainty that this data point will be classified to a label belonging to the rule after the post-processing procedure mentioned in Section 4.2.1.

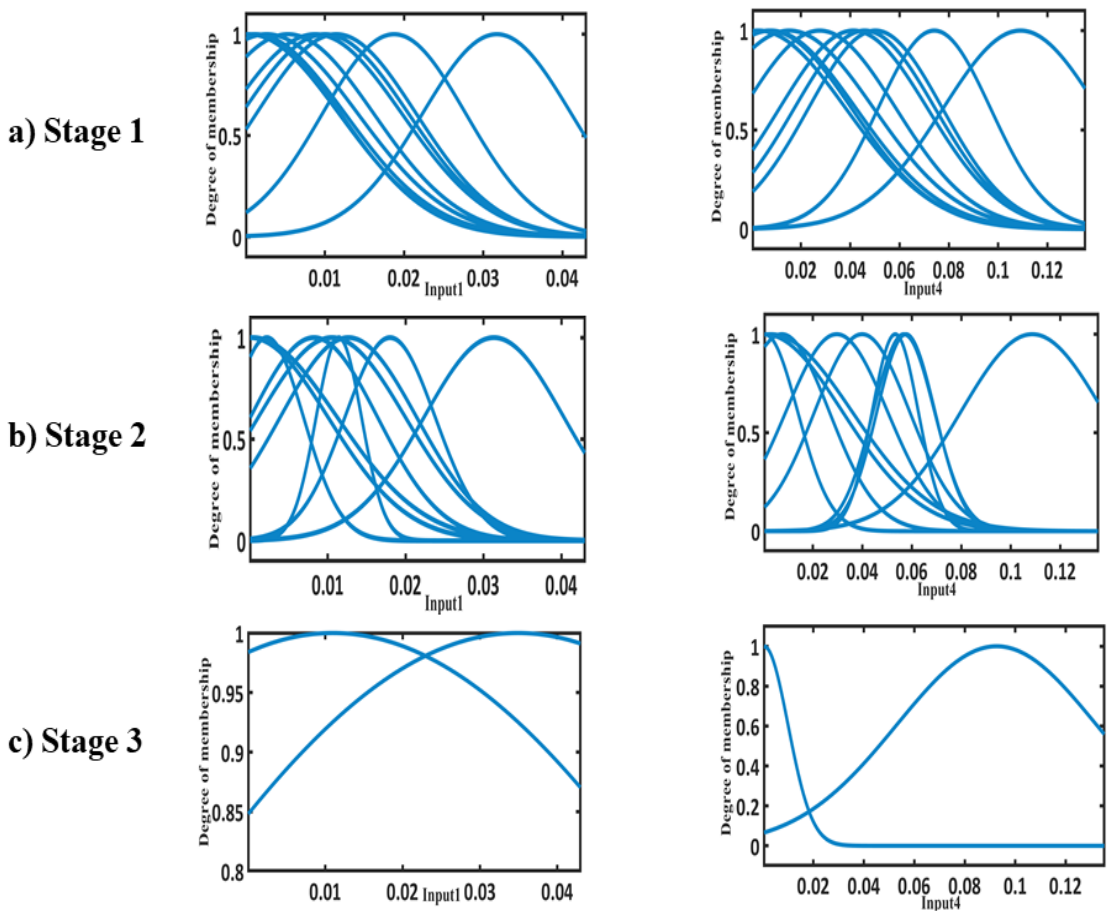

Figure 4.8. The membership functions of FRBS for the simplest classifier from a) Stage 1, b) Stage 2 and c) Stage 3 of IMOFM-C.

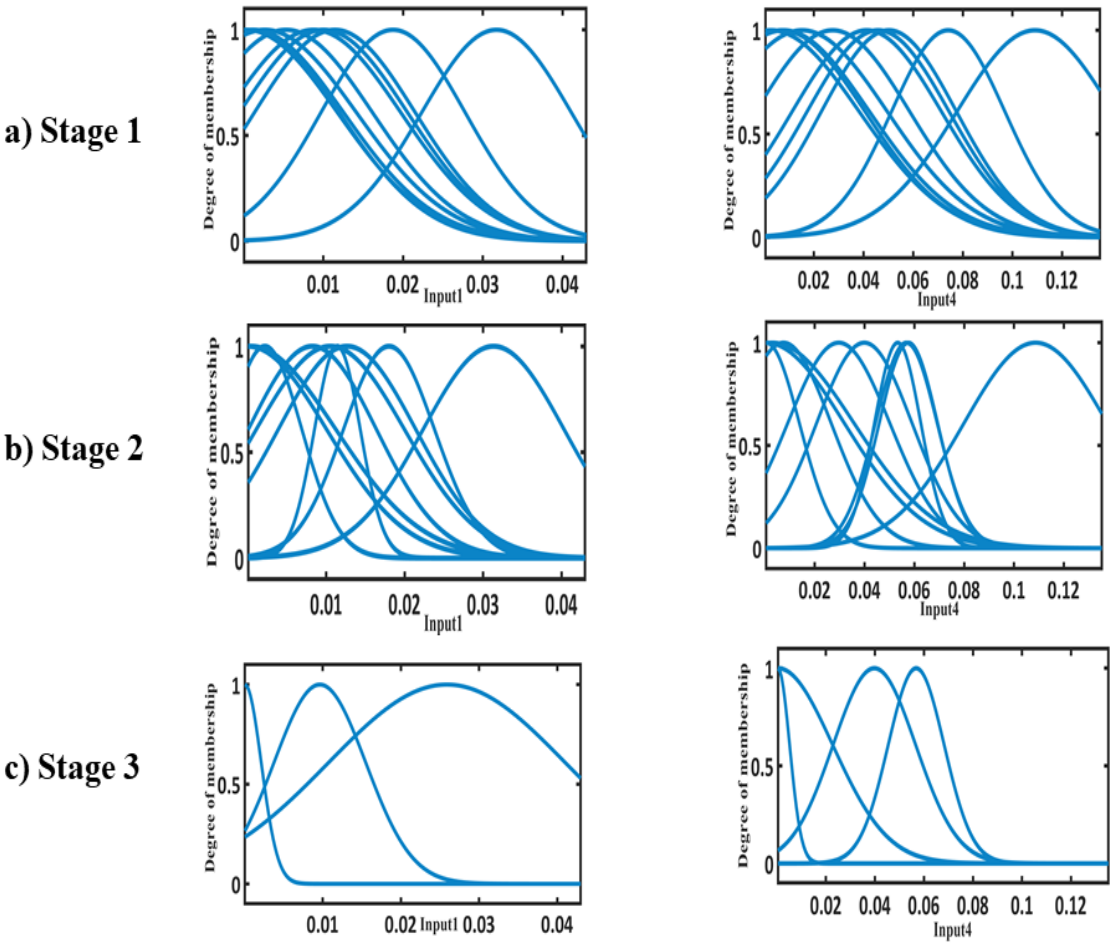

Figure 4.9. The membership functions of FRBS for the most complex classifier from a) Stage 1, b) Stage 2 and c) Stage 3.

In order to show the effectiveness of how the IMOFM-C classification algorithm improves interpretability in the third stage (see Section 4.2.1), the simplest classifier and the most complex classifier are chosen from the pareto of nondominated CF1 classifiers of subject 3 from one of the ten runs. The membership functions of the initial FRBSs from the first two stages and the simplified membership functions from the third stage of these two chosen classifiers (i.e. the simplest classifier and the most complex classifier) are shown in Figures 4.8 and 4.9 respectively using Input 1 and 4. As shown in both figures, in Stage 3 redundant fuzzy sets

have been removed and merged, resulting in a compact sets of membership functions for each input that is easier for interpretation.

4.4.5 Results on Multi-class Classification

In this section, the six binary classifiers trained with IMOFM, ANNs, ANFIS, CART and SVM algorithms obtained in Section 4.4.3 are aggregated using the aggregation methods described in Section 4.3. The performance of IMOFM on the multi-class problem on the testing sets are evaluated using the classification performance metrics defined in Eq. 4.9 and compared with the baseline algorithms.

To provide rigorous analysis, each algorithm was run for 10 times. Regarding the dataset, the testing sets provided at each one of the 10 runs for the six binary classifiers of each subject in Section 4.4.3 were put together and randomised. Finally, ten datasets consisting of all the four classes for each subject were created.

For the aggregation of the six binary classifiers based on the IMOFM mechanism, DDAG-Distance approach introduced in Section 4.3 is used. The same approach is applied to the six trained binary classifiers based on the ANFIS algorithm. The aggregation of binary classifiers based on ANNs, CART decision tree, and SVM were carried out using the original DDAG approach described in Section 4.1.2.

The performance of the IMOFM classification mechanism using four classification metrics, are compared with the baseline algorithms for Subjects 1-4, and is presented in Tables 4.24-4.27 respectively.

As it can be seen from the table of results, IMOFM perform significantly better on the multi-class problem comparing to the baseline algorithms with respect to the four classification performance metrics.

TABLE 4.24
COMPARSION OF MULTI-CLASS CLASSIFIERS- SUBJECT 1

Classifiers	Sensitivity	Specificity	Accuracy	Miss-points
IMOFM	**0.4501**	**0.8158**	**0.7314**	**169.2000**
ANNs	0.3795	0.7965	0.7014	188.1000
ANFIS	0.3147	0.7734	0.6681	209.1000
CART	0.3749	0.7981	0.6986	189.9000
SVM	0.3476	0.7866	0.6962	191.400

TABLE 4.25

COMPARSION OF MULTI-CLASS CLASSIFIERS- SUBJECT 2

Classifiers	Sensitivity	Specificity	Accuracy	Miss-points
IMOFM	**0.7040**	**0.8871**	**0.8400**	**124.8000**
ANNs	0.6497	0.8646	0.8081	149.7000
ANFIS	0.5980	0.8508	0.7881	165.3000
CART	0.6291	0.8660	0.8100	148.2000
SVM	0.5747	0.8549	0.7962	159

TABLE 4.26

COMPARSION OF MULTI-CLASS CLASSIFIERS- SUBJECT 3

Classifiers	Sensitivity	Specificity	Accuracy	Miss-points
IMOFM	**0.5525**	**0.8227**	**0.7534**	**344.7000**
ANNs	0.4963	0.8048	0.7285	379.5000
ANFIS	0.5106	0.8074	0.7326	373.8300
CART	0.5054	0.8048	0.7307	376.5000
SVM	0.4845	0.8067	0.7339	372

TABLE 4.27

COMPARSION OF MULTI-CLASS CLASSIFIERS- SUBJECT 4

Classifiers	Sensitivity	Specificity	Accuracy	Miss-points
IMOFM	**0.4885**	**0.8049**	**0.7206**	**179.4000**
ANNs	0.4442	0.7933	0.7061	188.7000
ANFIS	0.4148	0.7825	0.6864	201.3000
CART	0.4718	0.7996	0.7131	184.2000
SVM	0.4719	0.8019	0.7196	180

4.5 Conclusion

In this chapter, the IMOFM classification mechanism which was originally designed for regression problems is adapted for classification of multi-class SSVEP EEG data. The main contributions of this chapter are as follow.

1) The multi-objective modelling framework of IMOFM has been exploited to elicit a set of diverse and accurate FRBSs which can be used in designing effective ensemble classifiers in Chapter 5. Towards this, three different combinations of objective functions, including 'accuracy vs. diversity (PFC)', 'accuracy vs. diversity (NCL)' and 'accuracy vs. interpretability' were investigated. The results show that implementing the IMOFM classification mechanism with 'accuracy vs. interpretability' as the objective functions has not only encouraged the diversity of the solutions but also evolved significantly more accurate FRBSs comparing to the other two combinations. However, diversity measures could be used as a diversity preserving mechanism in the selection procedure of a search algorithm as will be discussed in the next chapter.

2) OAO decomposition-based approach is used to decompose the multi-class problem in to six binary classification problems. To aggregate the binary classifiers, the DDAG-Distance aggregation approach is proposed. Comparing to various baseline classification algorithms, such as ANFIS, ANNs, the CART decision tree, and SVM algorithm, the proposed IMOFM based multi-class classification algorithm presents superior performance in terms of both accuracy and interpretability.

Chapter 5

Ensemble Classifier based on Multi-objective Fuzzy Modelling for Classification of Multi-class EEG Data

In this chapter, a preference-based ensemble classification framework known as IMOFM-CP is developed based on the IMOFM classification mechanism introduced in Chapter 4.

In the proposed IMOFM-CP, the PAIA algorithm (see Chapter 4) is modified to incorporate the preference angle and reference information-based dominance (ar-dominance) (Yi et al. 2018). The rationale of such a modification and the main contributions of this chapter are as follows:

1. The modified PAIA algorithm increases the convergence speed of the population and reduces the number of solutions in the nonpreferred region (Yi et al. 2018). Therefore, classifiers in the nonpreferred region, i.e. the region that contains less accurate classification models, are automatically filtered out. On the contrary, classifiers in the preferred region, i.e. the region that contains more accurate classifiers, will undergo a more focused search, further improving their classification accuracy.

2. IMOFM-CP is designed to promote the diversity among the elicited classification models and fully utilise the power of each individual classification model as a member of a committee classifier (i.e. ensemble classifier). Aggregating a set of elicited classifiers leads to more accurate classification performance than that of the individual classifier or, in the

worst cases, equally good classification performance. Different aggregation methods are investigated in this chapter to identify the best one that works with IMOFM_CP.

3. IMOFM-CP is applied to both benchmark problems and the real-world multi-class EEG datasets. The results of IMOFM-CP are compared to those of IMOFM_C.

In the following, we first introduce the concept of ensemble classification, followed by a brief introduction to ar_MOEA, a novel multi-objective optimisation algorithm based on a preference-based dominance relation (Yi et al. 2018). Based on the preference-based dominance relation, the proposed preference based fuzzy ensemble classification mechanism is presented in Section 5.3. Finally, the experimental and comparison results are presented in Section 5.4.

5.1 Ensemble Classification Methods

Ensemble of classifiers are proven to have better generalisation capability than that of each individual classifier within the ensemble (Sagi and Rokach 2018). However, constructing a successful ensemble is not a trivial task. The two main components of designing an ensemble system are: a) generating a diverse and accurate set of classifiers, and b) aggregating the outputs of generated set of classifiers in a) that make up the ensemble in such a way that the correct decision are amplified and the incorrect ones are cancelled out (Polikar 2006). In this chapter the focuses are on investigating the approaches for generating diverse and accurate individual classifiers, the aggregation methods that deliver the best ensemble models for each binary classifier, and how to adapt the Decision Directed Acyclic Graph (DDAG) that has been introduced in Chapter 4 to combine many ensembles of binary classifiers into a multi-class classifier.

As discussed in Chapter 2, Section 2.5.4.2, there are different approaches in literature to construct a diversified and accurate ensemble.

The work in this chapter is based on forming a diverse yet accurate ensemble of classifiers is using evolutionary computational algorithm combined with machine learning based classifiers to optimise their parameters in a multi-objective optimisation manner. The evolutionary process of this approach will generate a population of classifiers which are different from each other and hence contribute to their diversity. An example of this approach can be seen in Xin Yao's work in 1998, in which a simultaneous evolution of ANN architectures and weights was introduced. Four linear combination methods were investigated for combining different individuals in the last generation into an ensemble (Xin Yao and Yong Liu 1998). Later, in 2004, Arjun Chandra and Xin Yao, introduced the DIVerse and ACcurate Ensemble learning algorithm (DIVACE), which formulates the ensemble learning of ANNs as a multi-objective problem explicitly (Chandra and Yao 2004). In DIVACE, the two objectives on which to optimise the performance of the ensemble are accuracy and diversity. For diversity, the negative correlation penalty function of NCL is used as one of the objectives. In 2006, the same authors, introduced a new diversity measure known as Pairwise Failure Crediting (PFC) to replace NCL in the DIVACE algorithm within a multi-objective formulation of the problem (Chandra and Yao 2006b). Unlike NCL that only compares the output of a classifier to the output of an ensemble, PFC measures the errors of individual classifiers on the training set with respect to all other classifiers in an ensemble.

In this chapter, the aim is to use the evolutionary computational algorithm combined with machine learning based classifiers to optimise not only the parameters of individua classifiers but also their structures to achieve a set of diverse and accurate classifiers.

5.2 MOEA based on Preference-based Dominance Relation

In all MOEAs, the aim is to facilitate the decision maker's (DM's) task by evolving better solutions. Integrating the preference information given by the DM into the search process is one of the approaches that have been recently investigated for achieving this aim (Ben Said, Bechikh, and Ghedira 2010). Integrating the preference information given by the DM into the EA makes the search process to be more focused and converged faster towards the region of interest (ROI) rather than searching the entire Pareto-optimal region. The preferences can be integrated in three ways (Ben Said, Bechikh, and Ghedira 2010): a) before optimisation (priori); b) during optimisation (interactively); c) after optimisation (posteriori). Interactive approaches have shown promising results comparing to the other two approaches, since in the interactive manner the search direction can be adjusted by exploiting knowledge and experience gained in the evolutionary process. ar-MOEA (Yi et al. 2018) a novel multi-objective optimisation algorithm based on a preference-based dominance relation is an example of an interactive based approach.

In IMOFM-CP, the proposed search process ar-PAIA is developed based on ar-MOEA approach (Yi et al. 2018) and Population Adaptive Immune Algorithm (PAIA) (J. Chen and Mahfouf 2006). In following the main principals of ar-MOEA is first provided, followed by the introduction of ar-PAIA in Section 5.3.

5.2.2 ar-MOEA

The ar-MOEA algorithm is aimed to create a stricter partial order among the nondominated solutions in order to force the Pareto front toward the ROI during the search process. This aim has been achieved by presenting a new variant of the Pareto dominance relation, known as preference angle and reference information-based dominance (ar-dominance). Before

explaining the ar-MOEA algorithm, there are several main concepts which are described below.

- Reference Point: Based on the literatures (Yi et al. 2018 ,Hadka and Reed 2013, Xiufen Zou et al. 2008, di Pierro, Khu, and Savi 2007, Jin and Sendhoff 2002, Köppen and Yoshida 2007), the dominance relation-based approaches are used to address the multi-objective optimisation problems with the help of preference information. Reference point is one of the preference information which is widely used in dominance relation-based approaches (Kalyanmoy Deb and Sundar 2006). Generally, the reference points can be prespecified in two particular locations (Yi et al. 2018): a) the feasible region, in which solutions near the reference point might mislead the population away from the Pareto front during iteration; b) the infeasible region, in which there are no elitist solutions to mislead the population away from the Pareto front.

- Weighted Euclidean Distance in ar-MOEA: Weighted Euclidean distance is calculated between the candidate solution and reference point r based on Eq. 5.1. This metric evaluates the degree of population convergence (Yi et al. 2018). Based on r-dominance method (Ben Said, Bechikh, and Ghedira 2010), the optimal solution falls between the original Pareto front and the reference point and the weighted Euclidean distance $D(r, x)$ is employed to determine the closeness of a certain solution to the reference point. In Eq. 5.1, m is the number of objectives, and $f_j(x)$ and r_j are the values of the objective function and the reference point in the jth dimension, and $j = 1,2, \dots, m$. w_j is the associated weight with the jth objective and, $w_j \in [0,1]$ and $\sum_{j=1}^{m} w_j = 1$. f_j^{max} is the upper bound and f_j^{min} is the lower bound of the jth objective.

$$D(r, x) = (\sqrt{\sum_{j=1}^{m} w_j \left(\frac{f_j(x) - r_j}{f_j^{max} - f_j^{min}}\right)^2} \qquad (5.1)$$

- Angle Information in ar-MOEA: The angle information $\theta(r,x)$ is calculated between the candidate solutions and the reference point. $\theta(r,x)$ is a decision factor for population diversity. Based on Eq.5.2, $\theta(r,x)$ is calculated between the reference point r and current objective vector $f(x)$ ·

$$\theta(r,x) = arccos\left(\frac{\sum_{j=1}^{m}|f_j(x)|\cdot|r_j|}{\sqrt{\sum_{j=1}^{m} f_j^2(x)}\sqrt{\sum_{j=1}^{m} r_j^2}}\right) \qquad (5.2)$$

- Adaptive Comprehensive Metric in ar-MOEA: To evaluate the ar-dominance relation of the solutions, first the following adaptive comprehensive metric is computed based on Eq.5.3 for all the solutions in the current population. The adaptive comprehensive metric is used to explicitly estimate the preference rank of each solution.

$$\varphi(r,x) = \left[\xi(t)\frac{\theta(r,x)}{\theta_{max}-\theta_{min}} + \left(1-\xi(t)\right)\frac{D(r,x)}{D_{max}-D_{min}}\right]\left(1+\frac{1}{e^m}\right) \quad (5.3)$$

Where m is the number of objectives $\xi(t)$ is the adaptive weight. t is the current generation number.θ_{max} and θ_{min} are the maximum and minimum values of calculated angel of the solutions from the reference point respectively. D_{max} and D_{min} represent the maximum and minimum values of the calculated distance of the solutions from the reference point, respectively.

- Adaptive Weight in ar-MOEA: Adaptive weight $\xi(t)$ is used to make adaptive trade-offs between the angle and distance information at different stages of the evolutionary process. In the ar-dominance relation, during the early generations, the priority is given to the distance information in order to enhance the selection pressure on convergence. In the later generations, the angel information between the solutions and the reference point is prioritised which results in enhanced selection pressure on diversity to explore the well-

distributed solutions. Therefore, $\xi(t)$ is adaptively changed during the iterative process based on Eq. 5.4.

$$\xi(t) = \xi_{min} + (\xi_{max} - \xi_{min})e^{-\left(1-\frac{t}{T}\right)} \qquad (5.4)$$

Where, T is the maximum number of generations, and ξ_{min} and ξ_{max} are the minimum and maximum weights, respectively.

- Adaptive threshold in ar-MOEA: Adaptive threshold is $\delta(t)$, which is used to control the distribution range of the solutions by constraining $\varphi(r, x)$. Adaptive threshold is computed based on Eq. 5.5 and can be adaptively changed during the iterative process. In the early generations, the constraints can be relaxed to retain more solutions, while in the later generations the constraints will be tightened to focus more on the quality of the solutions. The adaptive threshold is used to eliminate the suboptimal solutions and guide the population towards the ROI in later generations (Yi et al. 2018). The default range of δ is $[0,1]$. When $\delta = 0$, only one solution can be retained. When $\delta = 1$, all the solutions on Pareto front can be retained. In Eq.5.5, δ_{max} and δ_{min} represents the upper and lower bonds of the threshold and $\delta \in [\delta_{min}, \delta_{max}]$.

$$\delta(t) = (\delta_{max} - \frac{t}{T}(\delta_{max} - \delta_{min}) \quad (5.5)$$

- The ar-Dominance Relation in ar-MOEA: During the iterative process, the ar-dominance relation of the solutions can be evaluated as shown in ALGORITHM 5.1 (Yi et al. 2018).

In ALGORITHM 5.1 , assuming that P is the current population and p_1 and p_2, are the two given solutions (p_1, p_(2)∈P), solution p_1 is said to ar-dominant solution p_(2)(p_1 ≺_ar p_2), if it meets one of the following conditions.

ALGORITHM 5.1

AR-DOMINANCE RELATIONSHIP PROPOSED IN ar-MOEA

```
Pareto dominance comparison: between two given solution p_1 and p_2;
If (p_1 ≺ p_2) then
    p_1 ar-dominance p_2: (p_1 ≺_ar p_2);
    The elitist solution s is selected as: s ← p_1;
  else
    If (p_2 ≺ p_1) then
        p_2 ar-dominance p_1: (p_2 ≺_ar p_1);
        The elitist solution s is selected as: s ← p_2;
    End If
 End If
ar-dominance relation comparison: φ(r, p_1), φ(r, p_2) computed using Eq. 5.3;
If p_1 and p_2 are Pareto-equivalent && φ(r, p_1) − φ(r, p_2) < −δ then
    p_1 ar-dominance p_2: (p_1 ≺_ar p_2);
    The elitist solution s is selected as: s ← p_1;
 else
    If p_1 and p_2 are Pareto-equivalent && φ(r, p_2) − φ(r, p_1) < −δ then
        p_2 ar-dominance p_1: (p_2 ≺_ar p_1);
        The elitist solution s is selected as: s ← p_2;
    else
        p_1 and p_2 are ar-dominance equivalent;
        The elitist solution s: s ← randomly select (p_1, p_2);
    End if
 End if
```

1. In comparison of the Pareto dominance of the two solutions, p_1 dominates p_2 in the Pareto sense ($p_1 \prec p_2$).

2. If p_1 and p_2 are Pareto-equivalent, the comprehensive metric is calculated using $\varphi(r, p_1)$ and $\varphi(r, p_2)$ respectively. Therefore, $p_1$ar-dominant p_2, if $\varphi(r, p_1) - \varphi(r, p_2) < -\delta$.

- Archive Updating Strategy: The archive strategy have been widely used in different MOEAs such as: Pareto-Archive Evolution Strategy (PAES) (Knowles and Corne 1999), Strength Pareto EA (SPEA) (Eckart Zitzler and Zitzler 1999), Strength Pareto EA 2 (SPEA2) (Eckart

Zitzler et al. 2001) and Nondominated Sorting Genetic Algorithm II (NSGA II) (K. Deb et al. 2002). In the archive strategy, an external archive is maintained which contains a set of nondominated solutions that are found so far in the evolutionary process. The archive strategy is incorporated in MOEAs to avoid losing certain portion of the current nondominated front due to random effects (Eckart Zitzler et al. 2001). During the evolutionary process, an archive keeps updated in two steps: a) a solution in archive will be removed and replaced with a solution that has been found to dominate the archive solution; b) the current size of archive is checked; if the maximum archive size is exceeded, then an overcrowding measure will be used to remove the similar solutions in the archive in order to form a well-distributed archive of nondominated solutions. In the ar-MOEA approach, an archive updating strategy based on ar-dominance is proposed. The difference of the archive strategy used in this approach from the archive strategy that is used in NSGA II, PEAS, SPEA and SPEA2 is in the steps that are taken for updating an external archive. In ar-MOEA, instead of using Pareto domination comparison between the candidate solutions and the archive members, ar-dominance comparison is used to see if any of the candidate solutions ar-dominates a member in the archive. The ar-dominated member in the archive will be replaced. The size of the current archive A is obtained. If the size of the archive exceeds the maximum archive size n, then the members of the archive will be sorted in an ascending order based on their adaptive comprehensive metric values. Finally, the updated archive $\acute{A}$ will contain the first n members of the sorted archive A (Yi et al. 2018).

Incorporating the above concepts gives the ar-MOEA algorithm. A brief description of how ar-MOEA algorithm works is provided below.

Firstly, the initial population is randomly generated and evaluated according to the fitness function. Secondly, given the reference point r, the comprehensive metric based on Eq. 5.3 is calculated for each solution in the current population. In the meantime, the archive updating

strategy discussed above will be used to update archive A. Thirdly, the offspring population is produced after applying the simulated binary crossover and polynomial mutation. Finally, the ar-dominance relationship between the solutions in the current population and the offspring population is compared using ALGORITHM 5.1. Such comparison results in selecting the elitist solutions and archive A is updated using the discussed archive updating strategy.

The details of the ar-MOEA algorithm is shown in ALGORITHM 5.2 (Yi et al. 2018).

ALGORITHM 5.2

OVERVIEW OF AR-MOEA ALGORITHM

Generate initial random population $\boldsymbol{P(t)}$;

Evaluate the solutions in $\boldsymbol{P(gen)}$ based on the fitness function;

For $t = 1$ to T do:

  Calculate the comprehensive metric using Eq. 5.3: $\varphi\ (P(t), r)$;

  Producing offspring population $\boldsymbol{ch(t)^*}$:

  a) $ch(t) = crossover\ (P(t))$;

  b) $ch(t)^* = mutation\ (ch(t)$;

  Ar-dominance relationship comparison between $\boldsymbol{P(t)}$ and $\boldsymbol{ch(t)^*}$:

  ALGORITHM 5.1;

  Select the elitist solutions: $elitist(t) = ar - dominance\ (P(t), ch(t)^*)$;

  Update archive $\boldsymbol{A}$ with solutions from $\boldsymbol{elitist(t)}$: using Archive Updating Strategy

End

5.3 Preference based Fuzzy Ensemble Classification Mechanism

5.3.1 Rational for Improving IMOFM-C Approach

As discussed in Chapter 4, different objectives were tested to encourage the diversity as well as accuracy among the Pareto solutions. In IMOFM-C, diversity was achieved through

simplifying complexity of the fuzzy rule base. The adverse impact of doing so in the context of encouraging diversity is that potentially good models with complex structures can be compromised (i.e. deleted) in the early stage of evolution as a model with less complexity but more accuracy can dominate them. However, models with more complex structures may potentially evolve as the ones with more accuracy and at the same time promote diversity. Therefore, if the reference point is close to the most complex structure, and the adaptive threshold is set up such that it filters out those less accurate models, both diversity and accuracy can be improved. In light of this, IMOFM-C is adapted to incorporate the preference-based search concept in this chapter. The adapted IMOFM-CP aims to elicit a set of diversified models with sufficient accuracy for ensemble modelling.

5.3.2 Introduction of Proposed IMOFM-CP Mechanism

The structure of IMOFM-CP algorithm is the same as IMOFM-C mechanism which is described in Chapter 4. However, there is a difference in their search processes. In IMOFM-CP, the search algorithm is based on preference by incorporating the ar-dominance concept (i.e. ar-domince-PAIA). While in IMOFM-C, the search algorithm (PAIA) is not preference-based.

ar-dominance-PAIA is developed based on the ar-MOEA algorithm. The main difference between ar-dominance-PAIA and ar-MOEA are provided below.

- Weighted Euclidean Distance: In ar-dominance-PAIA, the weighted Euclidian distance is similar to $D(r, x)$ (see Eq.5.1) used in the ar-MOEA algorithm. However, in ar-dominance-PAIA, this metric is not using any normalisation as it is shown in Eq.5.6. The reason for using this metric without any normalisation is due to the preliminary experiments that were carried out using both normalised and non-normalised versions of this metric. The weighted Euclidean distance based on Eq. 5.1 could not find solutions in the preferred area when it

is incorporated in PAIA. While incorporating the non-normalised distance based on Eq. 5.6 in PAIA results in exploring the solutions in the ROI as it is expected. In order to distinct this metric from the one that is used in the ar-MOEA algorithm, $Dis(r, x)$ is defined in Eq. 5.6 to represent the weighted Euclidean distance in ar-dominance-PAIA. In Eq. 5.6, m is the number of objectives. $f_j(x)$ and r_j are the values of the objective function and the reference point in the jth dimension, and $j = 1,2, \dots, m$. Also, w_j is the associated weight with the jth objective. Where, $w_j \in [0,1]$ and $\sum_{j=1}^{m} w_j = 1$.

$$Dis(r, x) = (\sqrt{\sum_{j=1}^{m} w_j \, (f_j(x) - r_j)^2} \qquad (5.6)$$

- Angle Information: In ar-dominance-PAIA, the angle information is similar to $\theta(r, x)$ (see Eq. 5.2) in the ar-MOEA algorithm. However, carrying out the preliminary experiments and computing this metric based on Eq. 5.2 did not provide us with the results that was reported in (Yi et al. 2018). Therefore, in ar-dominance-PAIA, the angle information between the reference point r and the current objective vector $f(x)$is computed based on Eq.5.7, where instead of using the absolute values of $f_j(x)$ and r_j, their actual values have been used as this reflects the true angle information between $f_j(x)$ and r_j.

$$ang(r, x) = arccos\left(\frac{\sum_{j=1}^{m} f_j(x) \cdot r_j}{\sqrt{\sum_{j=1}^{m} f_j^2(x)} \sqrt{\sum_{j=1}^{m} r_j^2}}\right) \qquad (5.7)$$

In order to distinct this metric from the one that is used in the ar-MOEA algorithm, $ang(r, x)$ is used to represent the angle information in ar-dominance-PAIA.

- Adaptive Comprehensive Metric: In ar-dominance-PAIA, the adaptive comprehensive metric is calculated based on Eq.5.8 which is different from $\varphi(r, x)$ (see Eq.5.3) in ar-MOEA. In order to distinct this metric from the one that is used in the ar-MOEA algorithm,

$ang(r,x)$ is used to represent the angle information in ar-dominance-PAIA. In Eq.5.8, g represents the current generation number, and n represents the number of objectives.

$$ad\_c(r,x) = \left[\xi(g)\cdot(ang(r,x)) + \left(1-\xi(g)\right)\cdot(Dis(r,x))\right]\left(1+\frac{1}{e^n}\right) \quad (5.8)$$

- Adaptive Weight: In ar-dominance-PAIA, the adaptive weight matric ξ is calculated in the same way as in ar-MOEA using Eq.5.4. Therefore, the adaptive weight is presented by the same symbol (ξ) in both the ar-MOEA and ar-dominance-PAIA algorithms.

- Adaptive Threshold: In ar-dominance-PAIA, the adaptive threshold is calculated based on Eq.5.9 which is different from $\delta(t)$ (see Eq.5.5) in ar-MOEA. In order to distinct this metric from the one that is used in the ar-MOEA algorithm, $thr(g)$ is used to represent the adaptive threshold in ar-dominance-PAIA.

$$thr(g) = \left((thr_{max} - \left(\left(e^{-\left(1-\frac{g}{Gen}-(thr_{min})\cdot 1.5\right)}\right)^6\right))\cdot(thr_{max}-thr_{min})\right) \quad (5.9)$$

In ar-MOEA, the default range of thr is [0,1], where $thr = 0$ represents that only one solution can be retained, and $thr = 1$ indicates all the solutions can be retained. In the ar-dominance-PAIA algorithm, the maximum threshold thr_{max} and the minimum threshold thr_{min} is set by DM to guide the solutions toward the ROI. $thr \in [thr_{min}, thr_{max}]$ and it will be adaptively changed with respect to the stage of the evolution. In Eq.5.9, Gen represents the maximum number of generations.

- The ar-Dominance Relation: In ar-dominance-PAIA, during the iterative process, the ar-dominance relation of the solutions in the current population $pop(g)$ can be evaluated as follows: a) the dominated and nondominated solutions of the current population are identified as $nondom(g)$ and $dom(g)$ respectively; b) Computing adaptive weight $\xi(g)$ and adaptive threshold $thr(g)$ based on Eqs. 5.4 and 5.9 respectively; c) calculating the

comprehensive metric for all solutions in $pop(g)$ based on Eq. 5.8 which is represented by $ar\_all(pop(g))$; d) forming two vectors ar_A and ar_B, where ar_B is the inverse of ar_A; ar_A contains the comprehensive metric of nondominated solutions; the difference between ar_A and ar_B, ar_{diff}, will be calculated; e) the indices of those solutions that have higher comprehensive metric values with respect to other solutions, i.e. $ar_{diff} > thr(g)$ will be identified as Ind; and f) the nondominated solutions will be updated by removing those solutions with indices in Ind. The dominated solutions will be updated by adding those solutions removed from the nondominated set. The details of the ar-dominance relationship proposed in the ar-dominance-PAIA algorithm are provided in ALGORITHM 5.3

ALGORITHM 5.3.

AR-DOMINANCE RELATIONSHIP PROPOSED IN AR-DOMINANCE-PAIA

1 **Identifying the nondominated and dominated solutions in population $\boldsymbol{pop(g)}$:** $nondom(g)$, $dom(g)$;

2: **Calculating Adaptive weight:** $\xi(g)$ based on Eq. 5.4;

3: **Calculating Adaptive threshold:** $thr(g)$using Eq. 5.9;

4: **If** $thr(g) < thr_{min}$ **then**

5: $thr(g) = thr_{min}$;

6: **End**

7: **Calculating comprehensive metric for all solutions in $\boldsymbol{pop(g)}$:** $ar\_all(pop(g)) = ad\_c(r, pop(g))$ based on Eq. 5.8;

8: **Obtaining the comprehensive metric of nondominated solution from $\boldsymbol{ar\_all(pop(g))}$:** $ar\_nondom(g)$;

9: **Forming two matrixes as:** $ar_A = ar\_nondom(g)$; $ar_B = (ar_A)'$;

10: **Differences of two matrixes:** $ar_{diff} = (ar_A - ar_B)$;

11: **Obtaining $\boldsymbol{Ind}$:** $Ind = find\left(ar_{diff} > thr(g)\right)$;

12: **Updating** $nondom(g)$ and $dom(g)$;

- Archive Updating Strategy: In the ar-MOEA algorithm, an archive updating strategy is used to retain the nondominated solutions in an external storage with a fixed population size and to avoid overcrowding of the solutions in the external archive. In the proposed ar-dominance-PAIA, since the population is adaptive during the evolution, there is not an archive with a fixed size. The distribution of the population in ar-dominance-PAIA is maintained using the network suppression function (J. Chen and Mahfouf 2006). It is worth mentioning that the network suppression function will consider solutions in both $nondom(g)$ and $dom(g)$.

Incorporating the above principles into PAIA gives the preference-based PAIA – ar-dominace-PAIA.

ALGORITHM 5.4

OVERVIEW OF AR-DOMINANCE-PAIA ALGORITHM

```
1:   Generate initial random population pop(g) with size IN
2:   Evaluate the solutions in pop(g) using the ar-dominance relation: ALGORITHM
     5.3;
3:   For g = 1 to Gen do:
4:   |   Identify_Ab: This function is used to select a random Ab in the first Pareto front;
5:   |   Activation: This function is used to activate the other antibodies with the selected
     |   Ab;
6:   |   Clonal selection: this function selects the best individuals in the population in
     |   terms of their affinities and ar-dominance relations (refer to ALGORITHM 5.3);
7:   |   Clonal mutation: this function proliferates the selected solutions according to
     |   their affinities;
8    |   The mutated/edited offspring and their corresponding parents are mixed to-
     |   gether;
8:   |   Reselection:
8a.  |       Selecting all the nondominated solutions based on ALGORITHM 5.3:
     |       nondom(g);
8b.  |       If size ( nondom(g)) < IN then
     |       |  the next nondominated front is selected using ALGORITHM 5.3;
     |       End if
8c.  |       If size ( nondom(g)) > IN || g = Gen then
     |       | Network suppression function is applied to remove the similar solutions;
     |       End if
9    End
```

An overview of how ar-dominance-PAIA algorithm works is provided in ALGORITHM 5.4.

First, the initial population is randomly generated. and the solutions in the current population are evaluated based on their ar-dominance relation using ALGORITHM 5.3. Then, inside the iterative process the Identify Ab, Activation Clonal selection and Clonal mutation are used in the same way as those in the original PAIA (J. Chen and Mahfouf 2006).

Next, the mutated/edited offspring and their corresponding parents are mixed. Finally, the Reselection functions is used which contains 3 steps: a) all the nondominated solutions obtained based on ALGORITHM 5.3 are selected; b) the size of the nondominated solutions is checked; if it is smaller than the prespecified initial population size then the next-nondominated front which are obtained using ALGORITM 5.3 are selected; c) if the size of the nondominated solutions is greater than the prespecified initial population size or the maximum generation size is achieved, the network suppression function that is used in original PAIA (J. Chen and Mahfouf 2006) will be applied to remove similar solutions. As the result, a well-distributed near Pareto solutions within the ROI will be obtained.

Having described the new search algorithm known as ar-dominanace-PAIA that is used in the proposed IMOFM_CP mechanism, the rest of the IMOFM_CP mechanism remains the same as the IMOFM_C mechanism as described in Chapter 4.

5.3.3 Multi-Class Fuzzy Ensemble Classification Mechanism

As discussed in Section 5.1, an ensemble classifier normally leads to a higher classification accuracy (in particular in generalisation accuracy) than that of each individual member classifier (or its classification accuracy will be as good as the best individual member classifier). Therefore, the IMOFM-CP mechanism will be used to first generate a set of fuzzy classifiers within the ROI for each binary classification problem. It is worth mentioning that

the objective functions used in IMOFM-CP are complexity and accuracy as these two objectives promote the diversity among a set of generated fuzzy classifiers. As the result, for each binary classification problem (see Chapter 4), a set of diverse yet accurate fuzzy classifiers will be obtained. Average voting strategy will then be employed as the aggregation method to combine these individual member fuzzy classifiers in a bid to improve the classification accuracy (in particular for the testing dataset) of each binary classification problem. As a result, for each binary problem, there is a corresponding ensemble binary classifier.

For a multi-class classification problem, ensemble binary classifiers work in a similar way with the Decision Directed Acyclic Graph-Distance (DDAG-Distance) method developed in Chapter 4. Figure 5.1 illustrates the main steps of the proposed multi-class fuzzy ensemble classification mechanism.

As shown in Figure 5.1, first, a set of diverse yet accurate fuzzy classifiers will be obtained using IMOFM-CP. Second, ensemble classifiers for each binary classifier are formed containing the generated diverse and accurate individual classifiers in the ROI. Since DDAG-Distance requires information of the centres from a single fuzzy classifier, the best individual member classifier of each ensemble binary classifier is then identified. Next, the distance from a testing datapoint to the centres of the best individual member classifiers from each ensemble will be calculated. Finally, the minimum obtained distance indicates which ensemble binary classifier should be used in the DDAG-Distance method for a multi-class classification problem.

In the next section, the results of the proposed ar-dominance PAIA is first illustrated using two Multi-Objective Evolutionary Algorithm (MOEA) benchmark testing problems, viz. ZDT1 and ZDT3.

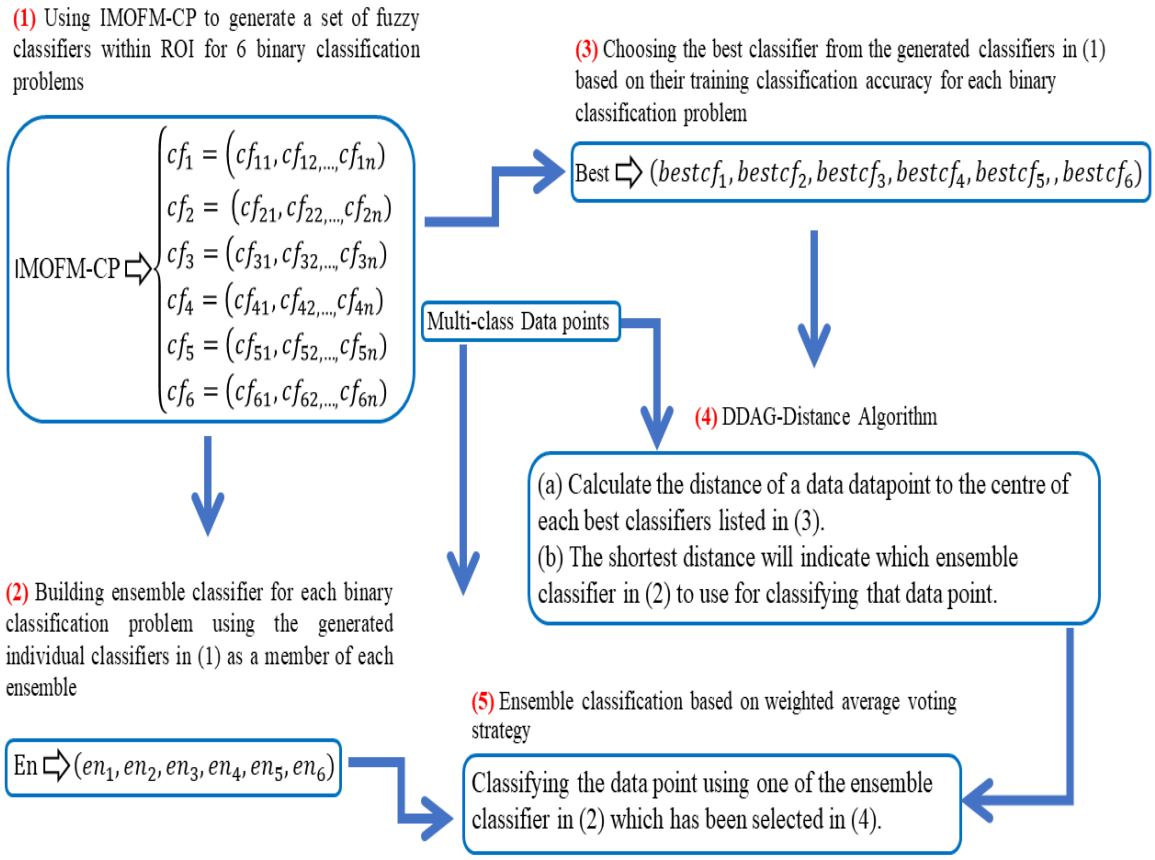

Figure 5.1 The Preference-based ensemble classification mechanism (IMOFM-CP) for a multi-class classification problem.

The performances of the proposed IMOFM_CP and its application to multi-class fuzzy ensemble classification problems have been tested and compared with the IMOFM_C mechanism and other baseline algorithms.

5.4 Experiments

5.4.1 The Results of ar-dominance PAIA on ZDT Benchmark Problems

In this section, ar-dominance PAIA algorithm is first tested on ZDT series problems (K. Deb 2001). For this experiment only ZDT1 and ZDT3 problems are employed to show the results

due to the space limitation. ZDT1 has 30 decision variables with a convex Pareto front. ZDT3 is a 30- decision variable problem with disconnected Pareto fronts. For more details on these test functions interested readers can refer to (K. Deb 2001).

Two experiments are designed to demonstrate how the reference point r, and the minimum and maximum thresholds thr_{min}, thr_{max} for computing the adaptive threshold can effectively adjust the ROI where desirable solutions reside.

- **Experiment 1: the effect of reference point.** Three different reference points are employed: a) the reference point with a focus on the top left part of the search space with $r = (1, 0.01)$; b) the reference point with a focus on the central part of the search space with $r = (0.5, 0.5)$; and c) the reference point with a focus on the bottom right part of the search space with $r = (0.01,1)$.

  The rest of parameters are set as follows. Initial population size 7, maximum generation size 500. For computing the adaptive threshold $thr(g)$ (see Eq.5.9), thr_{min} is set to 0.2 and thr_{max}, is set to 1 . To obtain the Adaptive weight (ξ) (see Eq. 5.4), ξ_{min} is equal to 0 and ξ_{max} is equal to 1.

  Figures 5.2 and 5.3 illustrate how ar-dominance-PAIA algorithm converges to three ROIs on ZDT1 and ZDT3 benchmark problems respectively.

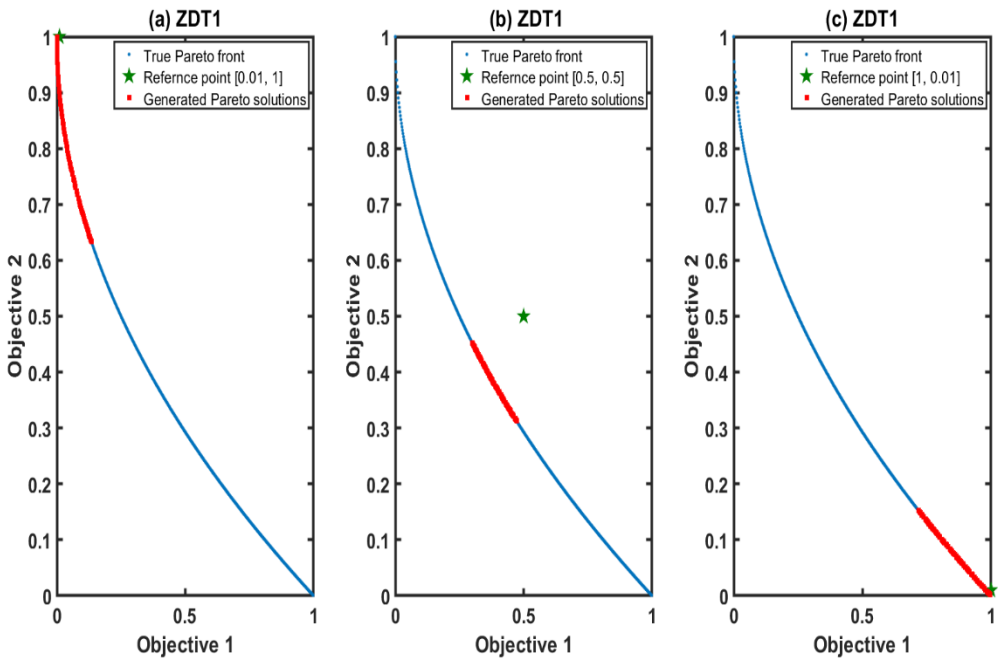

Figure 5.2 The performance of the ar-dominance-PAIA algorithm on ZDT1 problem with a) $r = (1, 0.01)$, b) $r = (0.5, 0.5)$, c) $r = (0.01,1)$.

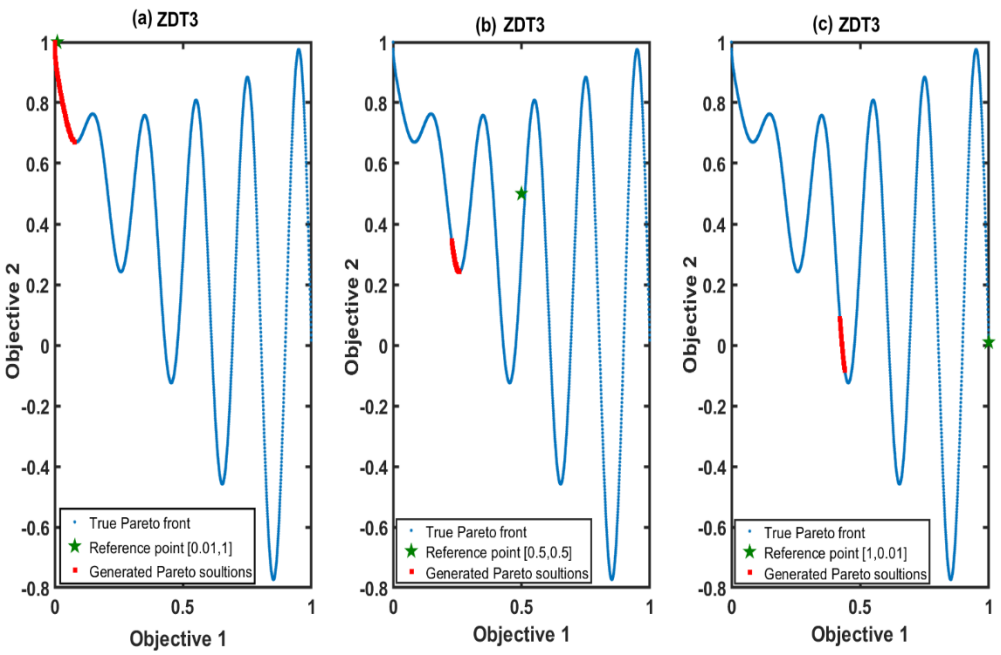

Figure 5.3 The performance of the ar-dominance-PAIA algorithm on ZDT3 problem with a) $r = (1, 0.01)$, b) $r = (0.5, 0.5)$, c) $r = (0.01,1)$.

As shown in Figures 5.2 and 5.3, by setting the reference point to $r = (1, 0.01)$, the search algorithm focuses on the top left part of the search space to find the most desirable solutions in that region of interest.

When setting the reference point to $r = (0.5, 0.5)$, the focus of the search algorithm changes to the central part of the search space Similarly, when setting the reference point to $r = (0.01,1)$, the focus of the search algorithm will be pushed further down to the bottom right of the search space and the desired solutions will be explored in that ROI.

- **Experiment 2: the effect of the minimum and maximum thresholds.** Two different combinations of thr_{min} and thr_{max} for computing the adaptive threshold are employed: a) $thr_{min} = 0.2$ and $thr_{max} = 1$; b) $thr_{min} = 0.98$ and $thr_{max} = 1$.

The rest of parameter settings are as follows. Initial population size 7, maximum generation size 500. To compute the Adaptive weight (ξ), ξ_{min} is equal to 0 and ξ_{max} is equal to 1. The reference point is fixed to $r = (1,0.2)$, i.e. the sesearch algorithm is expected to focus on the bottom right part of the search space. Different adaptive thresholds will lead to either wider or narrower approximate Pareto fronts.

Figures 5.4 and 5.5 illustrate how ar-dominance-PAIA algorithm performs by setting two different combinations of thr_{min} and thr_{max} on ZDT1 and ZDT3 benchmark problems respectively.

As shown in Figure 5.4 , by setting (a) $thr_{min} = 0.2$, and $thr_{max} = 1$, the search algorithm will focus on a smaller search space, eventually leading to a narrower approximate Pareto front. By increasing the value of thr_{min} to 0.98 (see Figure 5.5), the search algorithm encourages more search space to be explored, eventually leading to a wider approximate Pareto front.

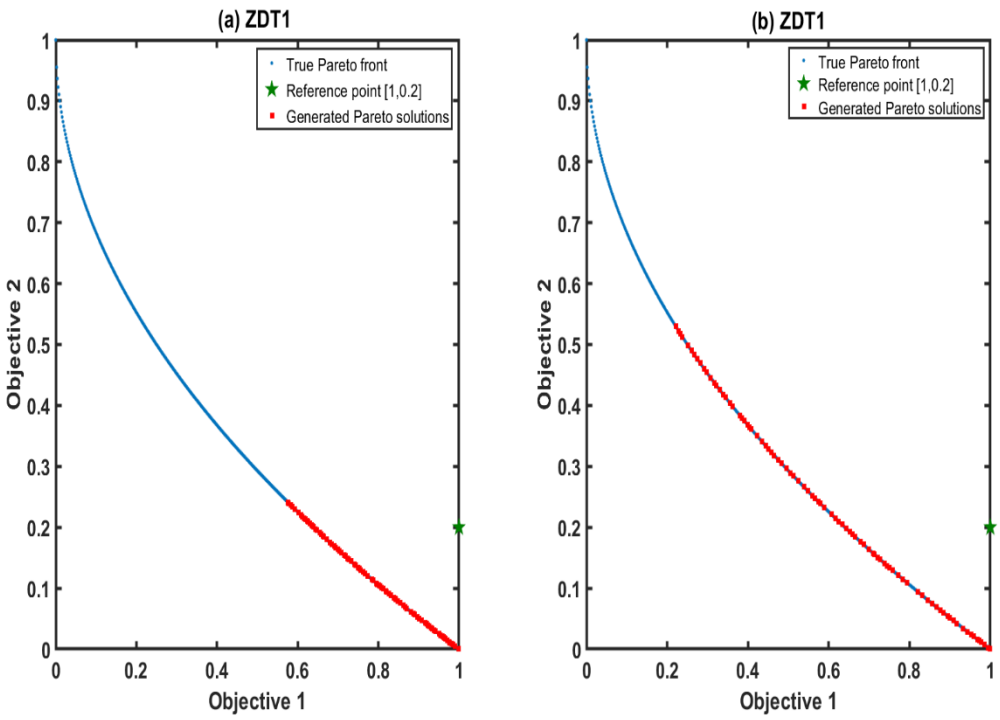

Figure 5.4 The performance of the ar-dominance-PAIA algorithm on ZDT1 problem with two different combinations of thr_{min} and thr_{max}: a) $(0.2, 1)$ and b) $(0.98, 1)$.

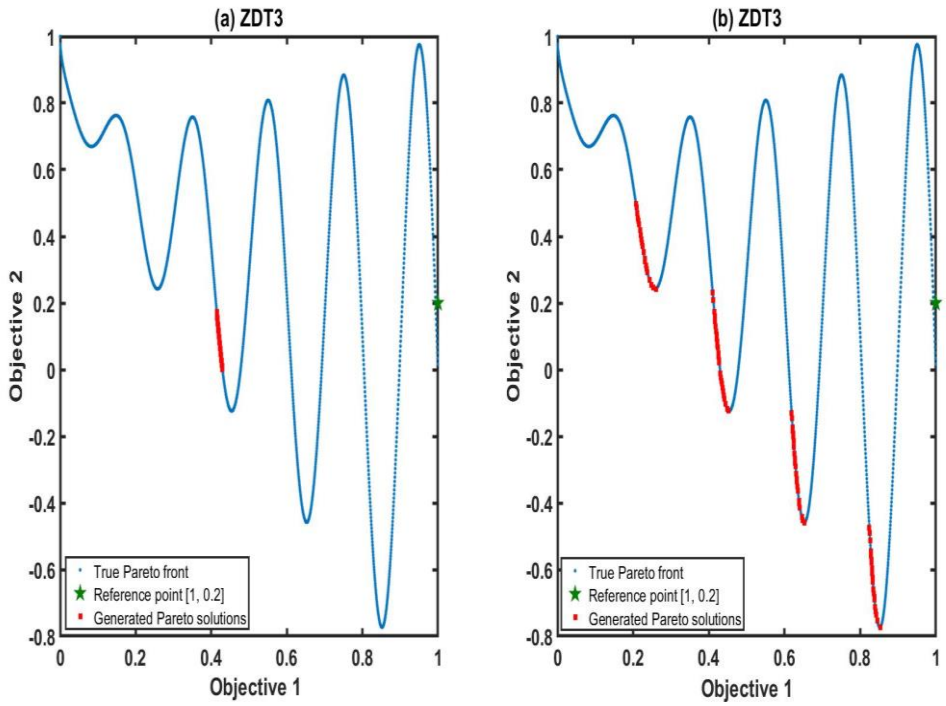

Figure 5.5 The performance of the ar-dominance-PAIA algorithm on ZDT3 problem with stwo different combinations of thr_{min} and thr_{max}: a) $(0.2, 1)$ and b) $(0.98, 1)$.

Setting $thr_{max} = 1$ enables the search algorithm to explore more search space in early iterations and then focus on smaller neighbourhoods for more focused exploitation as search approaches to the end. Obviously, setting thr_{max} to a smaller value may lead to a premature convergence as only smaller search space can be explored from the start of search.

5.4.2 Performance of IMOFM-CP on Binary-class EEG data

In this section, the suitable values for thr_{min}, and thr_{max} in calculating the adaptive threshold based on Eq.5.9 to obtain a set of diverse yet accurate (in particular for generalisation) binary classifiers are investigated. Towards this, eight different settings which represents eight different combinations of thr_{min} and thr_{max} are studied. As mentioned in Section 5.4.1, thr_{max} is fixed to 1, representing the whole Pareto front, to prevent premature convergence. Therefore, the focus of this section is first on finding an optimal value for thr_{min}. Increasing the value of thr_{min} will explore more search space and increase the covered area of the approximate Pareto front. However, increasing the value of thr_{min} to be close to 1 effectively reduces the search algorithm to the one without incorporating any preference. On the contrary, decreasing the value of thr_{min} allows a more focused search in the later stage of the search process, potentially leading to a better convergence in the ROI. In the following (see TABLE 5.1), thr_{min}, starts with the value of 0.30 (Setting 1) and is incrementally increased with a step size of 0.02 up to 0.4 (Setting 6). From Setting 6 onwards, the step size is changed to 0.1. It is worth noting that beyond $thr_{min} = 0.9$, the resulting approximate Pareto front is approaching to the whole approximate Pareto front that can be found by IMOFM-C. Therefore, only up to $thr_{min} = 0.8$ is included in this study.

TABLE 5.1
PARAMETER SETTING

Setting	(thr_{min}, thr_{max})	(ξ_{min}, ξ_{max})	Reference point
1	$(0.3, 1)$	$(0, 1)$	$(0.39, 0.2)$
2	$(0.32, 1)$	$(0, 1)$	$(0.39, 0.2)$
3	$(0.34, 1)$	$(0, 1)$	$(0.39, 0.2)$
4	$(0.36, 1)$	$(0, 1)$	$(0.39, 0.2)$
5	$(0.38, 1)$	$(0, 1)$	$(0.39, 0.2)$
6	$(0.4, 1)$	$(0, 1)$	$(0.39, 0.2)$
7	$(0.5, 1)$	$(0, 1)$	$(0.39, 0.2)$
8	$(0.6, 1)$	$(0, 1)$	$(0.39, 0.2)$
9	$(0.7, 1)$	$(0, 1)$	$(0.39, 0.2)$
10	$(0.8, 1)$	$(0, 1)$	$(0.39, 0.2)$

In all these settings the combination of ξ_{min} , and ξ_{max} for calculating the Adaptive weight based on Eq.5.4 is set to 0 and 1 respectively.

For the reference point, it is set to $(0.39, 0.2)$. Note, (0.39, 0.2) is in the normalised objective space. The decision for choosing the reference point as $(0.39, 0.2)$ is justified by our aim which is to find accurate enough binary classifiers with relatively simpler structures (i.e. with the less number of rules and similar fuzzy sets). This is inspired by the fact that binary classifiers with complex structures may lead to very high accuracy in training but generalise poorly in testing. On the contrary, binary classifiers with overly simplified structures may perform poorly in both training and testing. Therefore, the focus will be placed on searching for the classifiers which are not very complex but are still reliably accurate so that the generalisation capability of the ensemble classification can be increased as the result of such a focused and guided search.

The parameter configurations for the ten settings are listed in TABLE 5.1

Due to the space limit, in this section, results of IMOFM-CP with the above settings on two typical partitions for the classifier 4 (CF4) dataset for subject 2 are presented without the loss of generality. For this purpose, Partitions 1 and 2 are selected. It is worth mentioning that the CF4's dataset for subject 2 are the same dataset and partitions as the ones used in Chapter 4.

Partition 1 represents the case that even the binary classifier with the simplest structure still gives a satisfactory generalisation accuracy. While Partition 6 represents the case that those binary classifiers with simpler structures can lead to a very poor generalisation accuracy. In essence, these two partitions are employed for parameter analysis in order to derive the most reliable thr_{min} that can effectively discard those binary classifiers that are over trained (normally it comes with a more complex structure) and those ones perform poorly performed in generalisation (normally it comes with an overly simplified structure).

To provide a statistically reliable analysis, training of CF4 on the two portions with the IMOFM-CP mechanism for each setting has been run 10 times. The results are then averaged across 10 runs. For the comparison purpose, CF4 is also trained with IMOFM-C algorithm on the same two partitions for 10 times.

Since, both IMOFM-C and IMOFM-CP classification mechanisms are based on the multi-objective fuzzy approach, they evolve a Pareto front of solutions at each run. Therefore, two evaluation procedures for both IMOFM-C and IMOFM-CP have been employed as follows. 1) In the first evaluation procedure, in each run, from the nondominated FRBSs, the one with the lowest RMSE value is selected as the best classifier. The selected FRBS together with the post-processing procedure mentioned in Section 4.2.1 in Chapter 4 is then used to classify the training and testing sets. The performance of the selected classifier is evaluated using the classification performance measures including Accuracy, Sensitivity, Specificity and the total number of miss-classified data points as computed in Eq. 4.9 in Chapter 4. 2) In the second evaluation procedure, in each run, all the nondominated FRBSs are considered as members of an ensemble classifier and the Averaged voting strategy is used as the aggregation method. The result of the ensemble classification will go through the same post-processing procedure as in the first evaluation procedure. The performance of the ensemble classification will be

evaluated against the training and the testing sets using the same classification performance measures.

It is worth mentioning that in both IMOFM-C and IMOFM-CP classification mechanisms, the maximum generation number is set to 900, and the minimum and maximum number of rules are set to 2 and 10 respectively.

The performance of IMOFM-C and IMOFM-CP classification mechanisms on the training (TR) and the testing (TS) sets of Partitions 1 and 6 of CF4 are presented in TABLEs 5.2-5.5 respectively.

In TABLEs 5.3, and 5.5, IMOFM-CP classification mechanism are evaluated with the ten different settings listed in TABLE 5.1 for Partitions 1 and 6 respectively. For each setting, the performances of the IMOFM-CP classification mechanism using both evaluation procedures, Best classifier, and Ensemble classification are compared with those of the IMOFM-C classification mechanism in TABLEs 5.2 TABLE 5.4 respectively.

As it can be seen from the results for both classification mechanisms, the ensemble classification performs consistently better in all four performance classification metrics for testing comparing to the best classifier. Therefore, in the following, for the comparison between the IMOFM-C and IMOFM-CP classicisation mechanisms, the focus will be on the performances of IMOFM-CP using different thr_{min} settings.

In TABLE 5.3, the ensemble classification on the testing set for the IMOFM-CP classification mechanism with Settings 1 and 2 (i.e. Ensemble classification 1 and 2) performs consistently better than the ensemble classification on the testing set using the IMOFM-C classification mechanism (see TABLE 5.2). However, the performance of the ensemble classification for IMOFM-CP with Settings 3-6 were dropped comparing to the ensemble classification for

IMOFM-C. Then from Settings 7 to 10, the IMOFM-CP classification mechanism with the ensemble classification performs better comparing to the ensemble classification on IMOFM-C mechanism.

TABLE 5.2

IMOFM-C CLASSIFICATION MECHANSIM PERFOMANCE FOR THE BEST CLASSIFIER AND THE ENSEMBLE CLASSIFICATION ON PARTITION 1 OF THE CF4 CLASSIFIER DATASET FOR SUBJECT 2.

Classifiers IMOFM-C	**Accuracy (TR/TS)**	**Sensitivity (TR/TS)**	**Specificity (TR/TS)**	**Miss-points (TR/TS)**
Best classifier	0.8479/0.7468	0.8482/0.7475	0.8482/0.7475	35.90/20
Ensemble classification	0.8199/0.7532	0.8204/0.7540	0.8204/0.7540	42.50/19.50

TABLE 5.3

IMOFM-CP CLASSIFICATION MECHANSIM PERFOMANCE WITH TEN DIFFERENT SETTINGS FOR THE BEST CLASSIFIER AND THE ENSEMBLE CLASSIFICATION ON PARTITION 1 OF THE CF4 CLASSIFIER DATASET FOR SUBJECT 2.

Classifiers IMOFM-CP	**Accuracy (TR/TS)**	**Sensitivity (TR/TS)**	**Specificity (TR/TS)**	**Miss-points (TR/TS)**
Best classifier 1	0.8229/0.7430	0.8232/0.7437	0.8232/0.7437	41.80/20.30
Ensemble classification 1	0.8093/0.7608	0.8098.0.7615	0.8098.0.7615	45/18.90
Best classifier 2	0.8326/0.7392	0.8330/0.7399	0.8330/0.7399	39.50/20.60
Ensemble classification 2	0.8144/0.7570	0.8148/0.7576	0.8148/0.7576	43.80/19.20
Best classifier 3	0.8314/0.7380	0.8318/0.7387	0.8318/0.7387	39.80/20.70
Ensemble classification 3	0.8191/0.7468	0.8195/0.7477	0.8195/0.7477	42.70/20
Best classifier 4	0.8453/0.7291	0.8457/0.7296	0.8457/0.7296	36.50/21.40
Ensemble classification 4	0.8199/0.7405	0.8203/0.7412	0.8203/0.7412	42.50/20.50
Best classifier 5	0.8343/0.7367	0.8347/0.7373	0.8347/0.7373	39.10/20.80
Ensemble classification 5	0.8136/0.7443	0.8140/0.7450	0.8140/0.7450	44/20.20
Best classifier 6	0.84190.7405	0.8423/0.7412	0.8423/0.7412	37.30/20.50
Ensemble classification 6	0.8191/0.7456	0.8196/0.7464	0.8196/0.7464	42.70/20.10
Best classifier 7	0.8419/0.7443	0.8422/0.7448	0.8422/0.7448	37.30/20.20
Ensemble classification 7	0.8169/0.7608	0.8174/0.7617	0.8174/0.7617	43.20/18.90
Best classifier 8	0.8432/0.7405	0.8435/0.7411	0.8435/0.7411	37/20.50
Ensemble classification 8	0.8140/0.7608	0.8146/0.7618	0.8146/0.7618	43.90/18.90
Best classifier 9	0.8411/0.7494	0.8415/0.7501	0.8415/0.7501	37.50/19.80
Ensemble classification 9	0.8169/0.7582	0.8175/0.7592	0.8175/0.7592	43.20/19.10
Best classifier 10	0.8428/0.7342	0.8431/0.7347	0.8431/0.7347	37.10/21
Ensemble classification 10	0.8106/**0.7684**	0.8112/**0.7695**	0.8112/**0.7695**	44.70/**18.30**

TABLE 5.4

IMOFM-C CLASSIFICATION MECHANSIM PERFOMANCE FOR THE BEST CLASSIFIER AND THE ENSEMBLE CLASSIFICATION ON PARTITION 6 OF THE CF4 CLASSIFIER DATASET FOR SUBJECT 2.

Classifiers IMOFM-C	**Accuracy (TR/TS)**	**Sensitivity (TR/TS)**	**Specificity (TR/TS)**	**Miss-points (TR/TS)**
Best classifier	0.8453/0.7519	0.8456/0.7525	0.8456/0.7525	36.50/19.60
Ensemble classification	0.8280/0.7646	0.8284/0.7654	0.8284/0.7654	40.60/18.60

TABLE 5.5

IMOFM-CP CLASSIFICATION MECHANSIM PERFOMANCE WITH TEN DIFFERENT SETTINGS FOR THE BEST CLASSIFIER AND THE ENSEMBLE CLASSIFICATION ON PARTITION 6 OF THE CF4 CLASSIFIER DATASET FOR SUBJECT 2.

Classifiers IMOFM-CP	**Accuracy (TR/TS)**	**Sensitivity (TR/TS)**	**Specificity (TR/TS)**	**Miss-points (TR/TS)**
Best classifier 1	0.8479/0.7570	0.8443/0.7577	0.8443/0.7577	35.90/19.20
Ensemble classification 1	0.8326/0.7696	0.8331/0.7706	0.8331/0.7706	39.50/18.20
Best classifier 2	0.8475/0.7620	0.8479/0.7627	0.8479/0.7627	36/18.80
Ensemble classification 2	0.8314/0.7709	0.8319/0.7717	0.8319/0.7717	39.80/18.10
Best classifier 3	0.85/0.7582	0.8504/0.7588	0.8504/0.7588	35.40/19.10
Ensemble classification 3	0.8390/0.7759	0.8394/0.7767	0.8394/0.7767	38/17.70
Best classifier 4	0.8487/0.7620	0.8492/0.7628	0.8492/0.7628	35.70/18.80
Ensemble classification 4	0.8347/0.7646	0.8352/0.7654	0.8352/0.7654	39/18.60
Best classifier 5	0.8542/0.7582	0.8546/0.7590	0.8546/0.7590	34.40/19.10
Ensemble classification 5	0.8301/0.7722	0.8306/0.7730	0.8306/0.7730	40.10/18
Best classifier 6	0.8508/0.7544	0.8514/0.7552	0.8514/0.7552	35.20/19.40
Ensemble classification 6	0.8381/0.7709	0.8387/0.7718	0.8387/0.7718	38.20/18.10
Best classifier 7	0.8475/0.7582	0.8478/0.7589	0.8478/0.7589	36/19.10
Ensemble classification 7	0.8339/0.7709	0.8344/0.7717	0.8344/0.7717	39.20/18.10
Best classifier 8	0.8517/0.7506	0.8521/0.7513	0.8521/0.7513	35/19.70
Ensemble classification 8	0.8314/0.7734	0.8320/0.7742	0.8320/0.7742	39.80/17.90
Best classifier 9	0.8525/0.7658	0.8529/0.7664	0.8529/0.7664	34.80/18.50
Ensemble classification 9	0.8237/**0.7785**	0.8243/**0.7794**	0.8243/**0.7794**	41.60/**17.50**
Best classifier 10	0.8504/0.7620	0.8509/0.7626	0.8509/0.7626	35.30/18.80
Ensemble classification 10	0.8263/0.7722	0.8268/0.7729	0.8268/0.7729	41/18

For Partition 6 (TABLEs 5.4 and 5.5), the ensemble classification performances of the IMOFM-CP classification mechanisms in all the ten settings are consistently better than those of the ensemble classification with the IMOFM-C classification mechanism.

Combining the results presented in both TABLEs 5.3 and 5.5, it can be concluded that a too small value of thr_{min} may lead to a degradation in the generalisation performance of the

resulting ensemble classifier. Indeed, the best performing ensemble classifier for Partition 1 is Ensemble classification 10 where $thr_{min} = 0.8$. Although the best performing ensemble classifier for Partition 2 is Ensemble classification 3 (TABLE 5.5), Ensemble classification 10 still gives comparable generalisation accuracy. The improvement in both partitions is achieved as over trained and overly simplified classifiers are located outside the ROI. In light of the above, for the IMOFM-CP classification, it can be confirmed that, with carefully configured thr_{min}(along with the right choice of the reference point), IMOFM-CP can further improve the generalisation performance comparing to IMOFM-C.

Based on the results provided above, in order to choose a setting which works well across all the partitions for all the six classifiers (CF1, CF2, CF3, CF4, CF5, CF6), Setting 10 has been chosen for the implementation of the IMOFM-CP classification mechanism for the remaining experiments in this chapter.

The comparison of performances of the IMOFM-CP classification mechanism with the IMOFM-C classification mechanism for both best classifier and ensemble classification on all the 10 partitions of the training and testing datasets of the six classifiers for Subject 2 are presented in TABLEs 5.16 to 5.11 respectively. For each classification mechanism, each classifier is trained on 10 different partitions of each classifier dataset and their performances using the four classification performance metrics are averaged over 10 partitions and reported.

TABLE 5.6

COMPARIOSN OF CLASSIFICATION PERFORMANCES OF THE IMOFM-C AND IMOPFM-CP CLASSIFICATION MECHANISMES ON TRAING AND TESTING USING 10 PARTITIONS OF THE CF1 DATASET FROM SUBJECT 2

Classifiers	**Accuracy (TR/TS)**	**Sensitivity (TR/TS)**	**Specificity (TR/TS)**	**Miss-points (TR/TS)**
Best classifier IMOFM-C	0.9951/0.9582	0.9909/0.9371	0.9909/0.9371	0.8/2.30
Ensemble IMOFM-C	0.9933/0.9618	0.9882/0.9446	0.9882/0.9446	1.10/2.10
Best classifier IMOFM-CP	0.9951/0.9691	0.9909/0.9538	0.9909/0.9538	0.8/1.70
Ensemble IMOFM-CP	0.9951/**0.9709**	0.9909/**0.9592**	0.9909/**0.9592**	0.8/**1.60**

TABLE 5.7

COMPARIOSN OF CLASSIFICATION PERFORMANCES OF THE IMOFM-C AND IMOPFM-CP CLASSIFICATION MECHANISMES ON TRAING AND TESTING USING 10 PARTITIONS OF THE CF2 DATASET FROM SUBJECT 2

Classifiers	Accuracy (TR/TS)	Sensitivity (TR/TS)	Specificity (TR/TS)	Miss-points (TR/TS)
Best classifier IMOFM-C	0.9868/0.9315	0.9804/0.9115	0.9804/0.9115	2.10/3.70
Ensemble IMOFM-C	0.9801/0.9352	0.9743/0.9162	0.9743/0.9162	3.20/3.50
Best classifier IMOFM-CP	0.9851/0.9315	0.9791/0.9136	0.9791/0.9136	2.40/3.70
Ensemble IMOFM-CP	0.9776/**0.9389**	0.9726/**0.9208**	0.9726/**0.9208**	3.60/**3.30**

TABLE 5.8

COMPARIOSN OF CLASSIFICATION PERFORMANCES OF THE IMOFM-C AND IMOPFM-CP CLASSIFICATION MECHANISMES ON TRAING AND TESTING USING 10 PARTITIONS OF THE CF3 DATASET FROM SUBJECT 2.

Classifiers	Accuracy (TR/TS)	Sensitivity (TR/TS)	Specificity (TR/TS)	Miss-points (TR/TS)
Best classifier IMOFM-C	0.9974/0.9529	0.9955/0.9433	0.9955/0.9433	0.4/2.40
Ensemble IMOFM-C	0.9934/0.9510	0.9906/0.9400	0.9906/0.9400	1/2.50
Best classifier IMOFM-CP	0.9974/0.9529	0.9961/0.9414	0.9961/0.9414	0.4/2.40
Ensemble IMOFM-CP	0.9921/**0.9569**	0.9864/**0.9422**	0.9864/**0.9422**	1.20/**2.20**

As it can be seen from the results presented in TABLEs 5.6 to 5.11, the ensemble classification with the IMOFM-CP classification mechanism performs consistently better than the ensemble classification with the IMOFM-C classification mechanism for CF1, CF2, CF3 and C5. For CF4, the IMOFM-CP classification mechanism performs equally well as the IMOFM-C classification mechanism.

TABLE 5.9

COMPARIOSN OF CLASSIFICATION PERFORMANCES OF THE IMOFM-C AND IMOPFM-CP CLASSIFICATION MECHANISMES ON TRAING AND TESTING USING 10 PARTITIONS OF THE CF4 DATASET FROM SUBJECT 2

Classifiers	Accuracy (TR/TS)	Sensitivity (TR/TS)	Specificity (TR/TS)	Miss-points (TR/TS)
Best classifier IMOFM-C	0.8372/0.7544	0.8375/0.7548	0.8375/0.7548	38.40/19.40
Ensemble IMOFM-C	0.8182/**0.7722**	0.8189/**0.7779**	0.8189/**0.7779**	42.90/**17.60**
Best classifier IMOFM-CP	0.8398/0.7658	0.8401/0.7662	0.8401/0.7662	37.80/18.50
Ensemble IMOFM-CP	0.8220/**0.7722**	0.8226/**0.7779**	0.8226/**0.7779**	42/**17.60**

TABLE 5.10

COMPARIOSN OF CLASSIFICATION PERFORMANCES OF THE IMOFM-C AND IMOPFM-CP CLASSIFICATION MECHANISMES ON TRAING AND TESTING USING 10 PARTITIONS OF THE CF5 DATASET FROM SUBJECT 2

Classifiers	Accuracy (TR/TS)	Sensitivity (TR/TS)	Specificity (TR/TS)	Miss-points (TR/TS)
Best classifier IMOFM-C	0.8721/0.7974	0.8703/0.7951	0.8703/0.7951	28.90/15.40
Ensemble IMOFM-C	0.8544/0.8104	0.8518/0.8079	0.8518/0.8079	32.90/14.40
Best classifier IMOFM-CP	0.8783/0.8103	0.8772/0.7992	0.8772/0.7992	27.50/15.10
Ensemble IMOFM-CP	0.8562/**0.8211**	0.8535/**0.8183**	0.8535/**0.8183**	32/50/**13.60**

TABLE 5.11

COMPARIOSN OF CLASSIFICATION PERFORMANCES OF THE IMOFM-C AND IMOPFM-CP CLASSIFICATION MECHANISMES ON TRAING AND TESTING USING 10 PARTITIONS OF THE CF6 DATASET FROM SUBJECT 2.

Classifiers	Accuracy (TR/TS)	Sensitivity (TR/TS)	Specificity (TR/TS)	Miss-points (TR/TS)
Best classifier IMOFM-C	0.9285/0.8547	0.9272/0.8531	0.9272/0.8531	16/10.90
Ensemble IMOFM-C	0.9161/**0.8667**	0.9143/**08646**	0.9143/**08646**	18.80/**10**
Best classifier IMOFM-CP	0.9254/0.8387	0.9241/0.8354	0.9241/0.8354	16.70/12.10
Ensemble IMOFM-CP	0.9090/0.8507	0.9065/0.8485	0.9065/0.8485	20.60/11.20

5.4.3 Performance of IMOFM-CP on Multi-class EEG Data

In this section, the IMOFM-CP classification mechanism is implemented based on the framework defined in Section 5.3.3 for classification of multi-class EEG data from Subject 2. To provide a rigours analysis and comparison with the performances of other classification algorithms (IMOFM-C, ANNs, ANFIS, CART, and SVM) provided in TABLE 4.23 in Chapter 4, the IMOFM-CP classification mechanism has been run for 10 times on the same dataset used in other algorithms. For more details regarding the dataset used for this experiment please refer to Section 4.4.5 in Chapter 4.

The performances of the IMOFM-CP classification mechanism using the four classification metrics, are compared with those of IMOFM-C, ANNs, ANFIS, CART, and SVM and are presented in TABLE 5.12.

TABLE 5.12

COMPARSION OF MULTI-CLASS CLASSIFIERS FOR SUBJECT 2

Classifiers	Accuracy	Sensitivity	Specificity	Miss-points
IMOFM-CP	**0.8431**	**0.7119**	0.8392	**122.4000**
IMOFM-C	0.8400	0.7040	**0.8871**	124.8000
ANNs	0.8081	0.6497	0.8646	149.7000
ANFIS	0.7881	0.5980	0.8508	165.3000
CART	0.8100	0.6291	0.8660	148.2000
SVM	0.7962	0.5747	0.8549	159

As it be seen from the results, IMOFM-CP and IMOFM_C perform significantly better on the multi-class problem comparing to other classification algorithms with respect to three classification performance metrics.

5.5 Conclusion

In this chapter, a preference-based ensemble classification framework known as IMOFM-CP is developed based on the IMOFM-C classification mechanism introduced on Chapter 4. The main contributions of this chapter are as follow:

1) ar-dominance-PAIA search algorithm is designed to improve the convergence of the population in the ROI and reduces the number of solutions in the nonpreferred regions. Therefore, the classifiers in the nonpreferred region that contains less accurate or over trained classifiers are automatically discarded. The search algorithm will focus more on the ROI and the classifiers in that region will undergo a more focused search to further improve their classification accuracy.

2) IMOFM-CP classification mechanism is designed to promote the diversity among the classifiers and fully utilise the power of each individual classifier as a member of a committee classifier (i.e. ensemble classifier). Aggregating a set of elicited classifiers

leads to more accurate classification performance in generalisation than that of the individual classifier or, in the worst cases, equally good classification performance.

3) For multi-class classification problem, the ensemble of binary classifiers generated by the IMOFM-CP classification mechanism work with the Decision Directed Acyclic Graph-Distance (DDAG-Distance), which results in improved classification accuracy for multi-class classification comparing to all other baseline algorithms without using ensembles.

Chapter 6

Parameterisation of Neural Mass Models

In this chapter, first, a robust optimisation approach is introduced for parameterising a thalamic neural mass model that simulates brain oscillations such as those observed in EEG and local field potentials. Second, the TCT neural mass model will be parameterised with a revised objective function to simulate real SSVEP signal that has been recorded in Chapter 3.

Neural mass computational models of the thalamocortical brain circuitry are often used to mimic the meso-scale neuronal population behaviours such as those observed in electroencephalogram (EEG) and local field potentials (LFP)) (Liljenström 2012, Woldman and Terry 2015). The approach is based on physiological evidence of the fundamental role of feed-forward and feed–back connections between the thalamus and the cortex in generating and sustaining brain oscillations, also referred to as 'brain rhythms'. However, the main constraint of using neural mass models is the huge parameter space pertaining to these models; most of the model-based studies tune parameters by a trial-and-error method (Coyle et al. 2014, Bhattacharya, Coyle, and Maguire 2011), that produce a huge computational constraint in terms of time and efficiency. At the same time, biological plausibility of the parameter space is desirable to correlate model-based findings with physiological attributes, thus adding to model validation (Moran 2015) as well as to the translational value of the research.

The inherent nature of thalamocortical oscillations is highly non-linear and stochastic. Thus, it will not be too far from hypothesising that the neuronal and synaptic parameters and attributes

are in a constant state of change that may be major in the case of brain state changes such as from wakefulness to sleep; while local fluctuations within an 'acceptable upper and lower bounds' may underlie similar brain states under different environmental conditions. In fact, it may be hypothesised that minor variations in parameter values corresponding to similar brain states may underlie the well-known inter-individual differences observed in EEG. Towards this, our aim in this chapter is to use a biologically inspired optimisation algorithm for parameterising a neural mass model so that it oscillates with its first dominant frequency within the EEG alpha band (8-13 Hz). The parameterising has been carried out with respect to two scenarios: (1) a subject is in a relaxed but awake state with eyes closed; in this case the first dominant frequency is at 10 Hz without any subsequent harmonic frequencies; and (2) a subject is in an awake state and undergoes a rapid visual stimulation at 10 Hz; in this case the dominant frequency is at 10 Hz with the second dominant frequency at twice the stimulation frequency, as well as higher harmonics.

In following, first, a brief overview regarding the existing approaches for parameterising NMMs is provided. The thalamic module of thalamo-cortico-thalamic model is then presented, followed by the proposed robust optimisation algorithm that is used for searching the NMM's parameters that represent Scenario 1. The Experimental results for the parameterisation of the thalamic module are presented in Section 6.4. For Scenario 2, we first present an overview of the cortical module of the thalamo-cortico-thalamic model. Parameterising the thalamo-cortico-thalamic model under Scenario 2 is then introduced in Section 6.5, followed by the experimental results that are based on the real EEG data. Finally, conclusions are drawn in Section 6.6.

6.1 Overview of Parameterisation Approaches for Neural Mass Models

In Chapter 2, an overview of existing parameterisation approaches for NMMs are provided. Here, a more detailed overview of the most common parameterisation techniques such as those based on Kalman Filter and GA approaches are presented.

As mentioned in Chapter 2, One of the methods for parameterisation of NMMs is based on Kalman Filters (KF). Unscented Kalman Filter (UKF) and Cubature Kalman Filter (CKF) are examples of extension and generalisation to KF, which have been developed for parameterising NMMs. In following, a brief overview of three works in literature for parameterising NMMs based on KF techniques are presented.

Xian Liu's work based on UKF. In this work (X. Liu and Gao 2013), the proposed approach is based on UKF as an observer to estimate the unknown parameters of a neural mass model from noisy measurements. The purpose was to make the model simulate the EEG patterns, specifically the dynamical evolution during epilepsy seizure. They also developed a UKF based closed-loop control strategy to modulate the dynamics of the neural mass model in order to supress epileptiform spikes in the neural model. The neural mass model used in (Liu and Gao 2013) is based on Jansen and Rit's work (B H Jansen and Rit 1995) , which generates alpha rhythms based on the identified values of the parameters and several other parameters which are varied to mimic the epileptiform spikes.

The initial experiments carried out by setting all the parameters at their basal values and varying only one parameter referred to A, by increasing its value. The importance of estimating the unknown parameter A from noisy measurements has been confirmed and UKF was used for estimating A and reconstructing the output of the model. Due to the fact that measurements are

contaminated with noise, noise was added to the output and the prior knowledge about the noise characteristics was provided to UKF in order to have an accurate estimation.

An UKF based closed-loop control strategy was also developed which can supress the epileptiform spikes in the neural mass model by modulating the dynamics of the model. Based on this strategy, a feedback gain matrix is designed in the model, which by injecting the feedback to the identified hyperexcitable populations, the high amplitude epileptiform spikes can be turned into dynamics that reflects normal activities. It is shown that UKF performed well in estimating the unknown parameter and reconstructing the model output. Furthermore, the control energy used in Extended Kalman Filter (EKF) were compared with UKF and showed that the control energy needed in UKF is much less than the one needed in EKF. It is worth pointing out that this work was carried out in a very small scale with three coupled populations and just one unknown parameter. Therefore, further investigation is required to confirm the feasibility of such an approach for a larger and more complex model. The accuracy of the UKF algorithm is paramount which depends largely on the prior knowledge of the noise condition. Without such knowledge, the precision of UKF can drop dramatically.

Cuevas's work based on CKF: In this work (López-Cuevas et al. 2015), the parameterisation of the model is for the purpose of tackling epilepsy disease, which is one type of the brain disorders and is generated by abnormal and hypersynchronic activity in the brain. There is a serious type of seizure among others which is a long-lasting seizure and known as Status Epilepticus (SE). SE is defined as prolonged or recurrent seizures without full recovery between them. A neural mass model based on Jansen and Rit's model (B H Jansen and Rit 1995) is used to investigate whether there are any changes in the states and parameters of the model during the transition to SE. Two interacting areas were modelled: 1) the dentate gyrus (DG), which is a cortical region and part of a larger functional brain system known as the hippocampal formation, located between the entorhinal cortex and cornu ammonis areas; DG

plays an important role in processing information coming from the cortical areas and ultimately generating episodic memories; and 2) Cornu Ammonis (CA1) as it reliably reproduce different types of activities on the CA1 area, specifically, generating ictal, interictal and fast onset activities.

The acquired signal was first undergone a series of time-frequency analysis including 1) the empirical mode decomposition to separate frequency bands, 2) Hilbert-Huang transformation in order to obtain the time-frequency spectrum of the signals, and 3) Fourier transform to obtain the main frequency components of the selected segments of the signal. CKF is then used to estimate the states and parameters in an on-line fashion. According to the functionality of the biological system, there is a component known as homeostasis which is a property of internal self-regulated environment and maintains a stable system. Therefore, it plays an important role in the regulation of normal activities in the brain. It is believed that the relation between the internal parameters of the brain keeps the brain from experiencing and spreading seizures. As a result, in the case of pathological disorders, since homeostasis fails, the relation between the parameters is no longer stable. Therefore, the work is based on the hypothesis that when the key parameters of the model are estimated from a real pathological data, parameter relations should change during an ictal activity. The key parameters for estimation in this work include the excitation and inhibition gains and the parameter representing the strength connection for populations in two different areas.

Xian Liu's work based on a combined approach. In this work (X. Liu et al. 2014), a combined approach is proposed to estimate the states of a class of nonlinear NMMs with the amendment of the adaptive measurement noise. A model-based closed-loop control were also designed to quench the epileptiform spikes. The proposed approach is known as Fuzzy Adaptive UKF (FAUKF) and it is the combination of UKF and the fuzzy inference system (FIS) methods. UKF can achieve a very good performance if the information regarding the measurement noise

is known. However, this is in contrast to the real situation where the environment is changing constantly, and characteristics of noise cannot be accurately obtained due to its random nature. Therefore, the measurement noise can be adjusted adaptively through the FIS to enhance the accuracy compared to that of the UKF method alone. The NMM used in this work is also based on Jansen and Rit's model (Ben H. Jansen, Zouridakis, and Brandt 1993, B H Jansen and Rit 1995).

GA-based parameterisation approaches are promising techniques for parameterising NMMs. Compared to the manual fitting or least squares mean method (Zavaglia et al. 2008) and method based on kalman filters (X. Liu and Gao 2013). GAs offer the capability to capture different features of the observed EEG recordings in both frequency and time domains (F Cona et al. 2011). Owing to GAs' flexible framework and population-based search strategy, multi-objective GAs has been adopted to simultaneously capture different fitting requirements (Nevado-Holgado et al. 2014) rather than aggregating various requirements into a single objective (Nevado-Holgado et al. 2012). An overview of two existing works for parameterisation of NMMs based on GAs are provided in below.

Flippo Cona's work. The work presented in (Filippo Cona et al. 2009) is based on the works carried out by Melissa Zagavalia (Zavaglia et al. 2006, Zavaglia et al. 2008). Different interconnected Region of Interests (ROIs), including the thalamus, cingulated cortex, prefrontal cortex, primary motor area and supplementary motor area, are modelled in this work. In their previous works, they derived some connectivity patterns between three to five cortical regions by using the least squares minimisation. However, the least squares minimisation algorithm often converged to a suboptimal solution which leads to a poor fitting and obtaining values not bearing any physiological meanings. Therefore, in this work they proposed a new parameterisation method based on GAs.

The parameters involved in the parameterisation procedure include the reciprocal of time constants of excitatory synapses responsible for tuning the power peak frequencies and the connectivity strengths responsible for adjusting the amplitude of the power peaks. Other parameters are fixed at their biologically plausible values. The input to the model from external sources were simulated by the white Gaussian noise where the values for mean and variance were obtained manually for the first run of the algorithms, and they used the averaged optimised values for the next runs.

In the fitting procedure, they compared the model output with the real one in terms of their Power Spectral Density (PSDs). The aim of the objective function was to find a set of parameters that can minimise the difference between the real EEG and simulated signal. In order to improve the convergence time of the algorithm, they considered different objectives: 1) the mean square error ; 2) the similarity in the ratio between the local maxima and the local minima; 3) the position of the peaks; and 4) a single objective that aggregates all above objectives.

Holgado's work. The work presented in (Nevado-Holgado et al. 2012) is based on studying seizure and the underlying mechanisms that are responsible for the transition between different stages of seizure such as inter-ictal and ictal in the case of idiopathic generalised epilepsies (IGE). IGE is defined when a routine EEG recording of a patient shows the presence of 3 Hz spike and wave (SW) discharge, which is a waveform with a prominent spike and a slower wave component. In this work, the authors proposed a framework to estimate the parameters for the chosen NMM in order to make the model capable of generating an output that is matched to the specific features of the temporal evolution of the EEG recordings. In the modelling part of the work, they use a specific NMM to relate the model output to the EEG recordings from patients with the IGE problem. IGE is based on the interaction between two brain regions: the thalamus and cortex, which together play a vital role in generating SW rhythms. The model

consists of a mass of cortical neurons (e), cells in the reticular nucleus (r) and specific thalamic neurons (s).

The parameters involved in parameterisation include: 1) a purely cortical parameter Υ_e which is the ratio between the signal propagation velocity in the cortex and the typical cortical length scales; 2) a purely thalamic parameter A which is the inverse decay time of inhibition; and 3) α representing the behaviour of the thalamiccortical loop. The parameters described were chosen based on two reasons: 1) finding a minimal set of parameters that enables the model to generate the features of the recorded EEG data; and 2) the insights from past experimental and theoretical studies on IGE. Before fitting the model to real EEG data, they carried out several steps including filtering, segmentation and feature extraction from individual EEG cycles.

The proposed feature extraction algorithm in this work is based on the finite state machine, which is claimed to be robust to noise and artefacts. The feature extraction was carried out in the time-domain, in which every cycle presented a number of features including: 1) spikes, 2) the local minima between spikes (e.g. dips and inflection points) 3) the numbers, 4) their orders, and 5) their positions. All these features across a cycle represent the underlying physiological mechanisms to control the seizure evolution. Therefore, the aim is to find the optimal parameter values for each cycle to make the model reproduce similar features.

Regarding the parameterisation, a heuristic error function based on GAs was defined in order to identify a set of parameters for each cycle so that the aforementioned features of the recorded EEG can be reproduced. The parameter values can be adjusted by the heuristic error function. The error function depends on the similarity among the number, order and position of features between the seizure cycle and the modelled cycle. Considering the priority of the extracted features, the algorithm first attempts to find a combination of parameters that reproduce the same number of spikes in a cycle, then the ordering and finally the specific position of each

features inside a cycle. Therefore, a multi-objective problem has been aggregated to a single-objective optimisation problem using weighted-aggregation-based method. The results obtained from the data fitting procedure show the existence of more than one optimal parameter sets for each cycle. Therefore, a clustering algorithm has been used to choose the best parameter set.

6.2 Thalamo-Cortico-Thalamic Neural Mass Model

For the work presented in this chapter, we study the neural mass computational model of the thalamo-cortico-thalamic (TCT) circuitry in (Coyle et al. 2014, Bhattacharya, Coyle, and Maguire 2011), which originally used to simulate alpha rhythm slowing in Alzheimer's disease (AD). The TCT circuitry has been explained in more details in Chapter 2, Section 2.61.

Alpha rhythms are a prominent feature of the EEG occipital scalp electrode (the seat of the visual cortex) when a subject is in a relaxed but awake state with eyes closed. Furthermore, these oscillations are believed to be crucial for both conditions of visual attention and perception as well as for diminished cognition (Lorincz et al. 2014). More importantly, alpha band alterations often serve as EEG biomarkers in several disease conditions; for example longitudinal EEG studies show a shift of peak frequency within the alpha band (commonly known as 'slowing') as a definitive marker of AD (Coyle et al. 2014).

The visual pathway is by far the most widely studied thalamocortical pathway in experimental research (Sherman 2006).Thus, the parameterisation of the thalamic module in the TCT model has been based on physiological data on the synaptic structure and connectivity in the Lateral Geniculate Nucleus (LGN) of mammalian and rodent brains (Van Horn, Erişir, and Sherman 2000).It may be noted that most computational models of the thalamocortical circuitry ignore the feedforward inhibition to the thalamocortical relay cells from the thalamic interneurons.

This in spite the interneurons of the LGN receiving around 47% of their inputs from the retinal spiking neurons (Coyle et al. 2014). Furthermore, research suggests that the interneurons play a dominant role in efficient information transmission from the retina to the cortex (X. Wang, Sommer, and Hirsch 2011). To the best of our knowledge, the thalamic module in (Basabdatta Sen Bhattacharya, Coyle, and Maguire 2011b) looks into integrating thalamic inhibitory interneurons for the first time in NMMs of the thalamocortical circuitry.

6.3 Parameterisation of Thalamic Module in TCT Model

In this section, to illustrate the robust performance of the proposed parameterisation approach we ignore the cortical module of the TCT model and focus on optimising the parameter space of the thalamic module in the model. Hereafter in this work, we refer to the de-corticated thalamic module as the 'thalamic model'.

6.3.1 The Thalamic Neural Mass Model

The thalamic model is defined in Eqs. 6.1- 6.5 and consists of three cell populations: Thalamocortical Relay cells (TCR), Interneurons (IN) and Thalamic Reticular Nucleus (TRN).

TCR: $\ddot{y}_1 = a_1 H_e S\left(C_{tre} y_r + C_{tpe} y_4 - C_{tii} y_2 - C_{tni} y_3\right) - 2a_1 \dot{y}_1 - a_1^2 y_1$ (6.1)

IN: $\ddot{y}_2 = b_1 H_i S\left(C_{ire} y_r + C_{ipe} y_4 - C_{isi} y_2\right) - 2b_1 \dot{y}_2 - b_1^2 y_2$ (6.2)

TRN: $\ddot{y}_3 = b_1 H_i S\left(C_{nte} y_1 + C_{npe} y_4 - C_{nsi} y_3\right) - 2b_1 \dot{y}_3 - b_1^2 y_3$ (6.3)

Retinal: $\ddot{y}_\mathrm{r} = a_1 H_e P_1(t) - 2a_1 \dot{y}_\mathrm{r} - a_1^2 y_\mathrm{r}$ (6.4)

Sigmoid Function: $S(v) = \frac{2_{e0}}{1+e^{\upsilon(s_0 - v)}}$ (6.5)

H_e, is the strength of the excitatory (e) post-synaptic-potential (EPSP) and, H_i, is the strength of the inhibitory (i) post-synaptic-potential (IPSP); a_1 (b_1) is the inverse of the time constant of the excitatory (inhibitory) PSP; $P_1(t)$ is simulated by Gaussian white noise and represents the background firing activity of the retinal ganglion cells in the condition of eyes-closed, i.e. no sensory input (extrinsic input to TCR population from retinal cells); and $S(v)$ is a sigmoid function, where e_0 is the maximum firing rate of a neuronal population, s_0 is the resting membrane potential and v is the sigmoid steepness parameter; C_{xyz} are synaptic connectivity parameters with x representing the afferent population, y representing the efferent population and z representing either an excitatory or inhibitory synapse, and are defined in TABLE 6.1. (The reader may refer to (Lopes da Silva et al. 1974) for the other parameter values used in the equations).

TABLE 6.1

BASAL VALUES FOR THE SYNAPTIC CONNECTIVITY PARAMETERS USED IN Eqs. 6.1- 6.3. EACH THALAMO-CORTICAL CONECTIVITY PARAMETER IS A CONSTANT AND IS SOURCED FROM (Van Horn, Erişir, and Sherman 2000, Coyle et al. 2014).

Module	Afferent (to)	Efferent (from)	Connectivity Parameter	Value
Thalamic	TCR	Retinal	C_{tre}	7.1
Thalamic	TCR	IN	C_{tii}	15.45
Thalamic	TCR	TRN	C_{tni}	15.45
Thalamic	TCR	PY	C_{tpe}	62
Thalamic	IN	Retina	C_{ire}	47.4
Thalamic	IN	IN	C_{isi}	23.6
Thalamic	IN	PY	C_{ipe}	29
Thalamic	TRN	TCR	C_{nte}	35
Thalamic	TRN	TRN	C_{nsi}	15
Thalamic	TRN	PY	C_{npe}	50

The computational model of thalamic module is illustrated in Figure 6.1. Since the focus of parameterisation is only on thalamic module, the synaptic connectivity parameters between thalamic module and cortical module including, C_{tpe}, C_{ipe}, and C_{npe} are discarded.

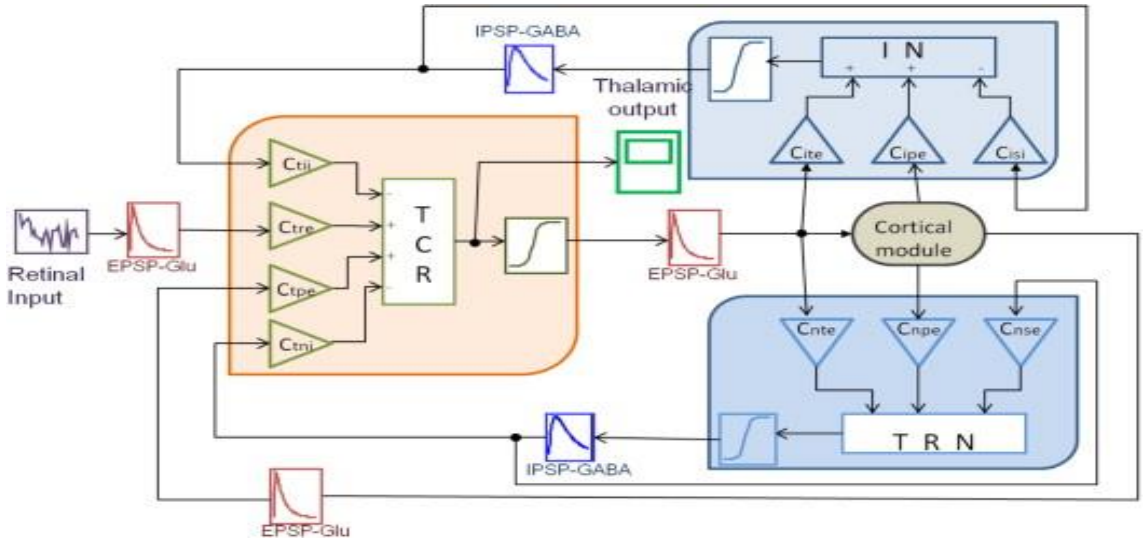

Figure 6.1 The computational model of the thalamic module (Bhattacharya, Coyle, and Maguire 2011).

The computational model of thalamic module is illustrated in Figure 6.1. Since the focus of parameterisation is only on thalamic module, the synaptic connectivity parameters between thalamic module and cortical module including, C_{tpe}, C_{ipe}, and C_{npe} are discarded.

6.3.2 Simulation and Signal Processing Methods for Thalamic Model

Model simulation is implemented using the 4th order Runge-Kutta ODE solver within the Simulink® environment in MATLAB. The total simulation time is 30 seconds at a sampling rate of 250 Hz. Each output vector thus obtained is bandpass filtered with a Butterworth filter of order 10 with a lower and upper cut-off frequency of 3 and 50 Hz respectively. The power spectral density analysis (PSDA) defined in Eq. 6.6 is performed in MATLAB using a Welch periodogram, with a Hamming window of segment length consisting of 125 data points and overlap of 50%. For the computational purpose, the output of the thalamic module generated in Simulink (simmodel) is passed through the PSDA (see Eq. 6.6). The outputs of the PSDA

are two vectors: a) the power density p, and b) its corresponding frequencies frq. The relative power RP is the normalised value of p with respect to the mean power and is computed using Eq. 6

$$(p, frq) = PSDA(simmodel(C_{xyz})) \tag{6.6}$$

$$RP = p./mean(p) \tag{6.7}$$

6.3.3 Robust Evolutionary Optimisation Approach for Parameterising C_{xyz}

Our objective here is to perform a rigorous search for the suitable values of C_{xyz} such that the dominant frequency model output lies within the alpha band (8 – 13 Hz). We expect that such an approach will provide us with a set of basal values for the connectivity parameter set C_{xyz} in the model. Towards this, we apply a standard search strategy using a single-objective GA. In this case, C_{xyz} are encoded as real values.The search space of C_{xyz} is created using ±30% across the basal values. The only exception is made for C_{tre}, which is varied ±15% due to its relatively smaller value. In order to have the overall power content within the alpha band, the standard GA searches for the maximum power $|mx|$ such that its corresponding frequency frq_{mx} lies within the alpha band. $|mx|$ and frq_{mx} are computed using Eq. 6.8. As the standard GA minimises the objective function, a minus sign is included in Eq. 6.8 for maximising $|mx|$.

$$(mx, frq_{mx}) = (-\max(RP)) \tag{6.8}$$

$$FP1 = \begin{cases} mx & if\ 8 \leq frq_{mx} \leq 13 \\ \frac{mx}{1.5+|8-frq_{mx}|} & if\ frq_{mx} < 8 \\ \frac{mx}{1.5+|13-frq_{mx}|} & if\ frq_{mx} > 13 \end{cases} \tag{6.9}$$

The objective function $FP1$ defined in Eq. 6.9 penalises the situation when $|mx|$ is outside of the alpha band.

As shown in Figure 6.2, the search process stagnates as variations in the extrinsic input is changing the objective landscape of $FP1$, leading to an entirely new search process for every generation of the GA. Thus, we observe that the standard optimisation algorithm fails, which may be attributed to the stochastic nature of the extrinsic input to the model.

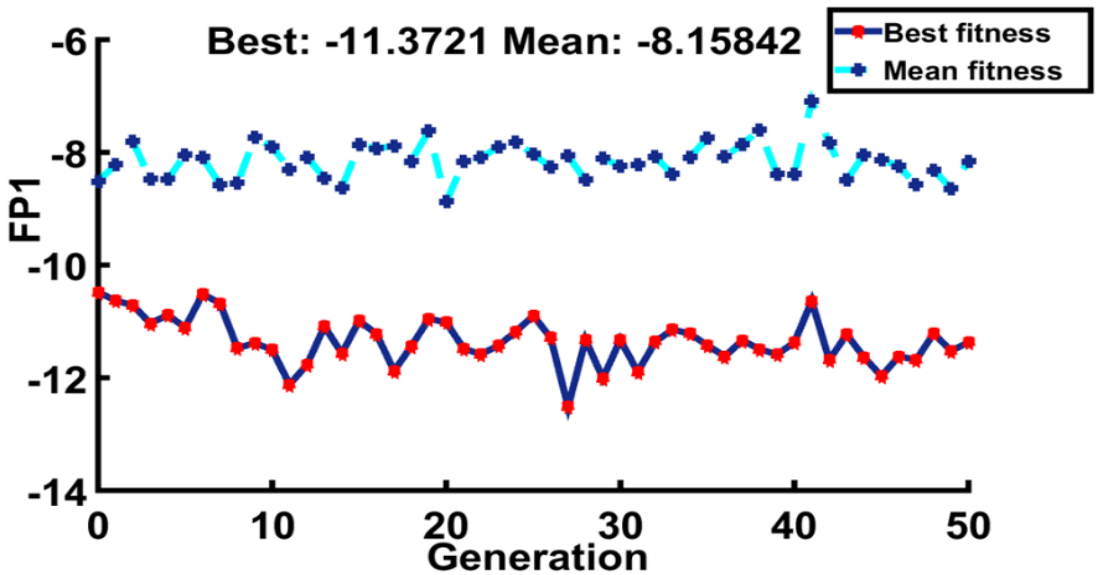

Figure 6.2 The evolutionary curve of the GA using $FP1$.

In light of this and drawn upon more recent concepts in robust optimisation, an improved robust parameterising approach is proposed here to address the stochastic issue due to the extrinsic input, and to identify reliable regions to account for small variation in C_{xyz} (e.g. due to environmental/inter-individual differences).

Here, 'robustness' is defined as the likelihood of $|mx|$ within the alpha band given a certain set of C_{xyz}. To enhance the robustness of the chosen C_{xyz}, two revised objective functions based on Eq. 6.9 are introduced, followed by a clustering algorithm and finally the overall parameterising framework.

Revised Objective Functions Considering External Uncertainty. $FP2$ is defined in Eq. 6.10 as the averaged power spectra peak calculated over n randomly generated extrinsic inputs to take into account the stochastic nature of the extrinsic inputs. Here, $n = 35$ and is an empirical number derived from the experiment.

$$FP2 = \sum_{i=1}^{n} FP1_i / n \tag{6.10}$$

Compared with Eq. 6.9 which maximises the peak with respect to only one fixed Gaussian white noise, the effect of Eq. 6.10 is to maximise the robustness indirectly. To directly maximise robustness, a counter which calculates how many times $|mx|$ is located outside of the alpha band is defined in Eq. 6.11. $FP3$ is further developed in Eq. 6.12 to take into account robustness and $FP2$ so that the constraint devised as the penalty function in Eq. 6.9 is also incorporated.

$$cr = \sum_{i=1}^{n} counter(i)$$

$$counter(i) = \begin{cases} 0 \; if \; 8 \leq frq_{mx}(i) \leq 13 \\ 1 \; otherwise \end{cases} \tag{6.11}$$

$$FP3 = cr + 1/|FP2| \tag{6.12}$$

Clustering and Robust Regions Considering Minor Variations in Parameter Values. As discussed in previous part, instead of a fixed set of C_{xyz}, it often makes more sense to have robust regions which can accommodate small variations in C_{xyz}.To this aim, solutions from each generation of the search will be filtered through a predefined threshold based on their objective values. The collection of filtered solutions represents the solution set to C_{xyz}. An evolutionary clustering algorithm-G3Kmeans (J. Chen et al. 2014) is then applied to group these solutions. G3Kmeans is the hybridisation of a real valued GA with a new crossover (recombination) based on the parent-centric principal (J. Chen et al. 2014) and the *K*-means algorithm, resulting in a less sensitive clustering to the initial settings. To identify the number of

clusters, subtractive clustering in MATLAB is first applied with the cluster radii set to the default value of 0.5.

The obtained clusters after G3Kmeans, provide the upper and lower bounds for each region. To further investigate whether these regions are robust for all possible C_{xyz} within them, m random C_{xyz} within the identified bounds are sampled. The robustness RG of each region is then computed using Eq. 6.13, where, cr is defined in Eq. 6.11.

$$RG = \frac{\sum_{j=1}^{m} cr_j}{m} \tag{6.13}$$

The Overall Robust Parameterising Framework. The overall robust parameterising framework is outlined in Figure 6.3.

```
1:  Set the decision variables (C_xyz) in the GA
2:  For gen = 1 to gen_max do:
3:      GA calls the objective function
4:      Save C_xyz
5:  End
6:  Collect all the good C_xyz with a threshold
7:  Subclust to give the number of candidate regions
8:  G3Kmeans to obtain the potential robust regions
9:  Randomly generate other C_xyz within the robust regions
10: Investigate robustness of the identified regions
```

Figure 6.3 The overall robust parameterising framework.

In Line 1, C_{xyz} are encoded as real values. In Line 3, the GA calls one of the objective functions defined in Eqs. 6.10 and 6.12. The collected c_{xyz} solutions from the GA will then be passed to the clustering algorithms (Lines 6-8). RG will then be calculated for each of these regions to evaluate their robustness (Lines 9-10).

6.4 Computational Results for Parameterising Thalamic Model

In this Section the results for parameterising Thalamic model of TCT circuitry using the proposed robust optimisation approach are provided.

6.4.1 Performances of the Robust Optimisation Approach

The experiments for the proposed robust optimisation approach are implemented using the GA toolbox in MATLAB®. To derive statistically reliable results, the experiments were carried out 10 times respectively for $FP2$ and $FP3$, with 25 generations and the population size of 10. Other parameters are set as default. The evolutionary curves of the robust optimisation approach using $FP2$ and $FP3$ are illustrated in Figure 6.4 (a) and (b) respectively. A significant improvement and better convergence in terms of the mean fitness were achieved compared to the one using $FP1$.

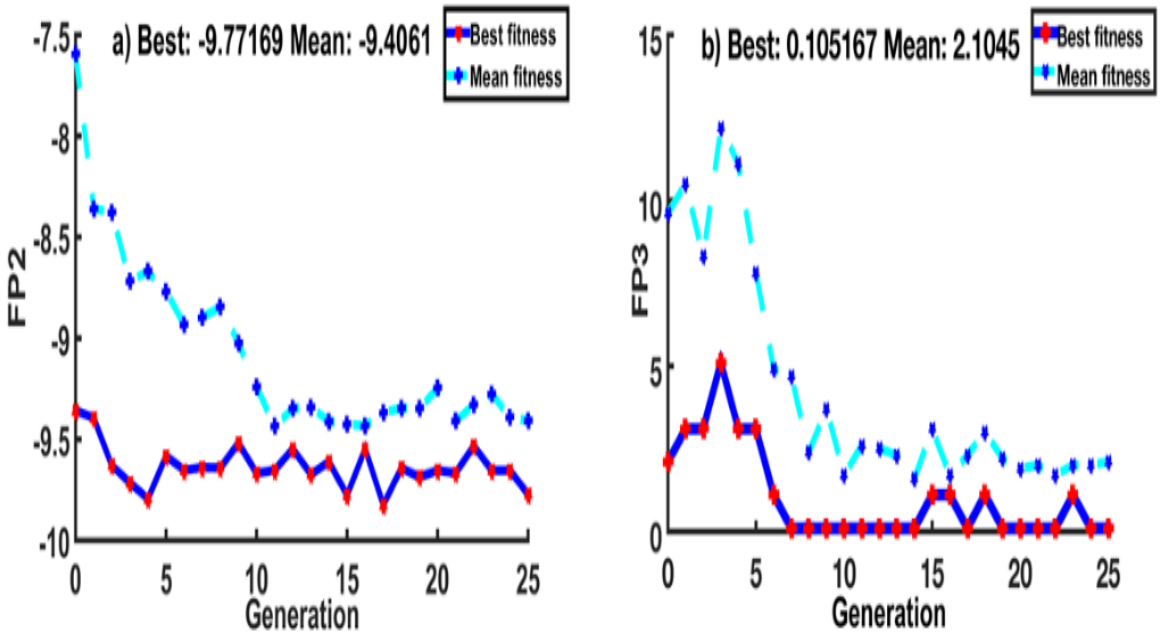

Figure 6.4 The evolutionary curves of the GA using a) $FP2$ and b) $FP3$.

Furthermore, Figure 6.4 (b) shows that $FP3$ leads to better convergence than $FP2$ due to the direct optimisation of the robustness. Therefore, the results reported in the following are based on $FP3$. It is worth noting that the negative signs in Figure 6.2 and Figure 6.4 (a) are due to the reason mentioned in Section 6.3.3

6.4.2 The Identified Regions using Clustering

Following Line 6 in Figure 6.3, 260 solutions (C_{xyz}) obtained from the robust optimisation approach are collected. Subtractive clustering indicates there are 3 clusters representing three potential regions (R1-R3). Figure 6.5 shows these regions and their centres after G3Kmeans clustering using the dimensions C_{nsi},C_{tre} and C_{nte}.

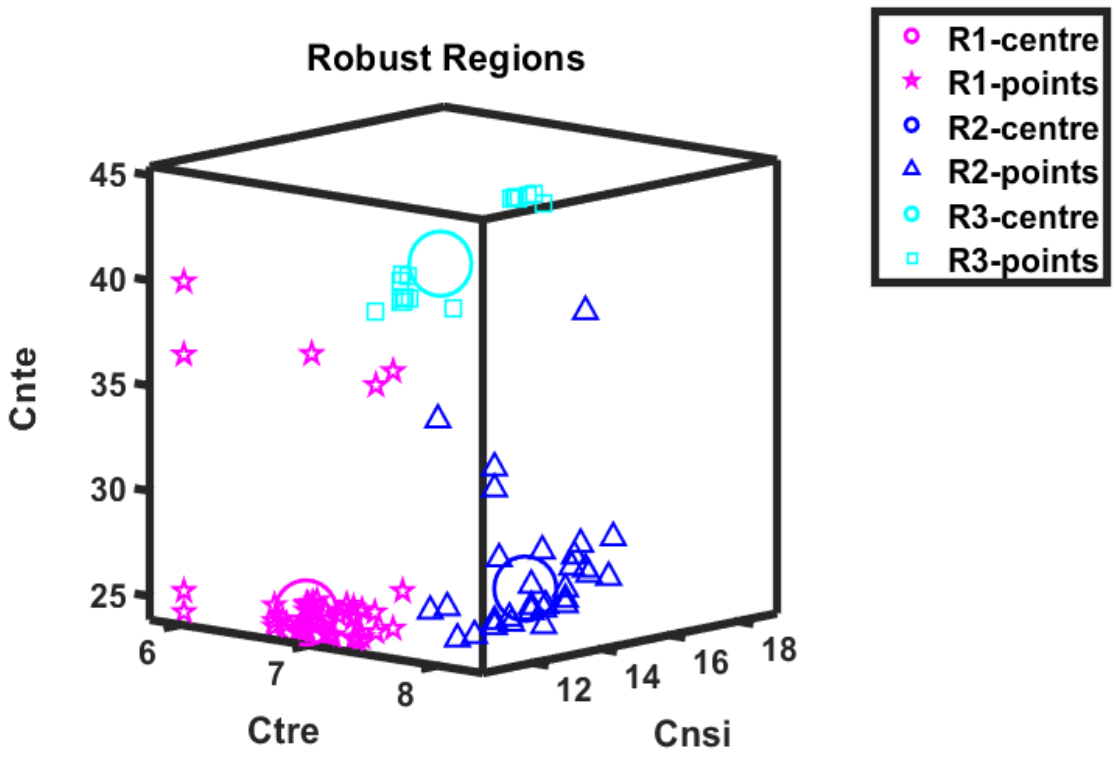

Figure 6.5 Three potential regions identified using the revised objective function $FP3$.

6.4.3 Robustness of the Identified Regions

To further investigate robustness of the regions identified in Section 3.2, their corresponding RG is calculated using Eq. 6.13 following Lines 9-10 in Figure 6.3, with $m = 50$. For comparison purpose, $m = 40$ are used for the basal values. TABLE 2 summarises the comparison results based on the basal values (BV) and the identified regions using $FP3$. Results indicate that the identified regions are robust not only at solutions obtained using $FP3$ (e.g. R1 via $FP3$), but also at randomly generated solutions within these regions (e.g. Random R1). Results also reveal that multiple robust regions may exist, which are more robust than the region around the basal values.

TABLE 6.2

COMPARISON OF ROBUSTNESS OF THE SYNAPTIC CONNECTIVITY PARAMER SETS C_{xyz}.

Methods	**Synaptic Connectivity Parameters [C_{ire}, C_{isi}, C_{tre}, C_{tii}, C_{tni}, C_{nte}, C_{nsi}]**	**Robustness (Mean RG)**
Basal values (BV)	[47.4,23.6,7.1,15.45,15.45,35,15]	13.00
C_{xyz} out of the identified R	**[53,28.15,9,17.8,20,45,18]**	14.00
Lower Band	**Upper Band**	
R around BV	**[42.66,21.24,7.09,13.90,13.90,31.5,13.5] [42.66,21.24,7.09,13.90,13.90,31.5,13.5]**	**9.50**
R1 via $FP3$	**[33.18,16.52,6.04,10.87,10.87,24.5,10.5] [35.18,25.66,7.53,15.12,19.83,40.2,13.13]**	**0.57**
R2 via $FP3$	**[33.18,16.52,6.31,10.87,10.87.24.5,11] [40.32,27.24,8.04,16.46,19.83,36.73,19.13]**	**0.32**
R3 via $FP3$	**[33.18,16.52,7.53,10.87,10.87,40.2,10.5] [33.68,17.52,8.09,12.87,19.44,44.75,14.35]**	**0.24**
Random R1	**[33.18,16.52,6.04,10.87,10.87,24.5,10.5] [35.18,25.66,7.53,15.12,19.83,40.20,13.13]**	**4 .26**
Random R2	**[33.18, 16.52,6.31,10.87,10.87.24.5,11] [40.32,27.24,8.04,16.46,19.83,36.73,19.13]**	**6.08**
Random R3	**[33.18,16.52,7.53,10.87,10.87,40.2,10.5] [33.68,17.52,8.09,12.87,19.44,44.75,14.35]**	**2.10**

In Figure 6.6 (a)-(f), C_{xyz} are randomly generated from R1 via $FP3$, Random R1, BV, $\pm 10\%$ across BV and those outside of any identified regions, and their corresponding power spectrum are plotted. Figure 6.6 (a)-(b) show that the aim of maximising the likelihood of $|mx|$ towards the alpha band is achieved. Indeed, the resulted dominant frequency is consistently within the upper-alpha band which is in line with the EEG of a young healthy adult. This conclusion holds true for R2 and R3 although they are not included here for brevity. Figure 6.6 (c)-(d) and (f)

show that the maximum power content is frequently (up to 40%) below the alpha band. For $\boldsymbol{C}_{xyz}$ outside of any identified regions, the chance of below the alpha band (as shown in Figure 6.6 (e)) is up to 30%.

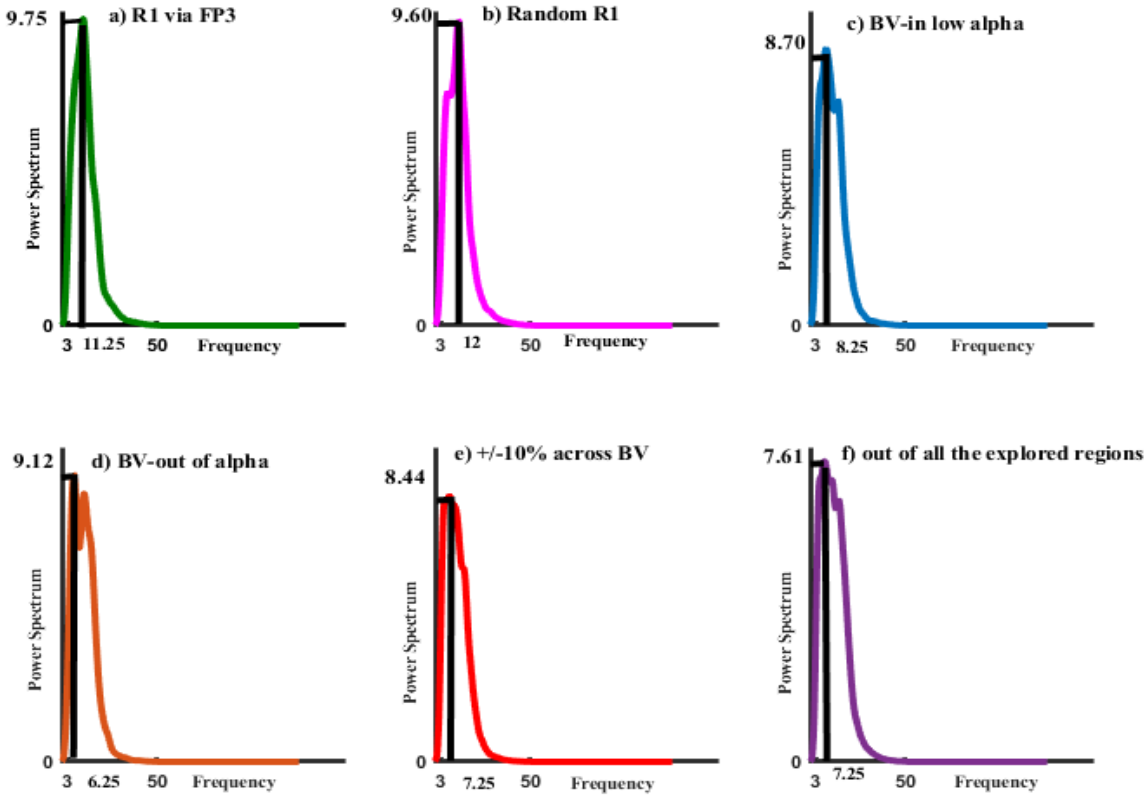

Figure 6.6 (a)-(f) The power spectrum corresponding to different C_{xyz} within different regions.

6.5 Data Fitting Procedure for TCT Model with Real EEG Data

The results from parameterisation of the thalamic model as part of the TCT circuitry in Section 6.4 using the robust optimisation approach suggest that multiple robust and distinct parameter regions exist. The results also show that the parameterised thalamic model is able to produce the consistent dominant frequency of oscillation within the alpha band. The cortical model also plays an important role in the TCT circuitry in generating the SSVEP. Therefore, in this section,

we will extend the parameterisation approach to the whole TCT circuitry in order to parameterise the TCT model with respect to the real EEG SSVEP data. As the result, for each segment of the SSVEP signal (See Chapters 4 and 5), a set of corresponding connectivity parameters of the TCT model will be found such that the TCT model can faithfully simulate the observed SSVEP signal. Therefore, the training and testing sets for the proposed IMOFM classification mechanism in Chapter 4 can be replaced with the corresponding connectivity parameters. As these parameters are biologically plausible and governed by the functions of the cortical and thalamic modules, using them as the features for classification algorithms may improve the classification accuracy. Toward this, we will first provide a brief overview of the Cortical module as part of the TCT circuitry, followed by the experimental results using the resting EEG signal and SSVEP signal at 10 Hz. Hereafter in this work, we refer to the disconnected cortical module from thalamic module as the 'cortical model'.

6.5.1 Overview of the Cortical Model

The cortical model is based on the works in (Wendling et al. 2002, Zavaglia et al. 2006) and consists of four cell populations: Pyramid cells (PY), Excitatory Inter-Neurons (eIN), slow inhibitory Inter-Neurons (sIN) and fast inhibitory Inter-Neurons (fIN). The cortical model is defined in Eqs. 6.14-6.18.

PY: $\ddot{y}_4 = a_1 H_e S\left(C_{pce} y_c + C_{pte} y_1 + C_{pxe} y_5 - C_{pli} y_6 - C_{pfi} y_7\right) - 2a_1 \dot{y}_4 - a_1^2 y_4$ (6.14)

eIN: $\ddot{y}_5 = a_1 H_e S\left(C_{xpe} y_4\right) - 2a_1 \dot{y}_5 - a_1^2 y_5$ (6.15)

sIN: $\ddot{y}_6 = b_{il} H_{il} S\left(C_{lpe} y_4\right) - 2b_{il} \dot{y}_6 - b_{il}^2 y_6$ (6.16)

fIN: $\ddot{y}_7 = b_{if} H_{if} S\left(C_{fpe} y_4 - C_{fli} y_6\right) - 2b_{if} \dot{y}_7 - b_{if}^2 y_7$ (6.17)

Cortico-cortical: $\ddot{y}_{\mathrm{c}} = a_1 H_e P_2(t) - 2a_1 \dot{y}_{\mathrm{c}} - a_1^2 y_{\mathrm{c}}$ (6.18)

It is worth mentioning that these equations and the thalamic module equations defined in Eqs. 6.1-6.4 form the complete TCT circuitry.

$b_{il/f}$,, is the inverse of the time constant of the inhibitory PSP generated by sIN (l) or fIN (f) cell population, respectively; $P_2(t)$ is simulated by a Gaussian white noise and represents extrinsic input to the cortical model from the neighbouring cortical regions. The sigmoid function, $S(v)$ is the same as defined in Eq. 6.5. For the other parameter definitions and values used in the above equations, please refer to Section 6.3.1 and the work by (Lopes da Silva et al. 1974). The computational model of the cortical module is illustrated in Figure 6.7 and the synaptic connectivity parameters, C_{xyz} for the cortical module are defined in TABLE 6.3 .

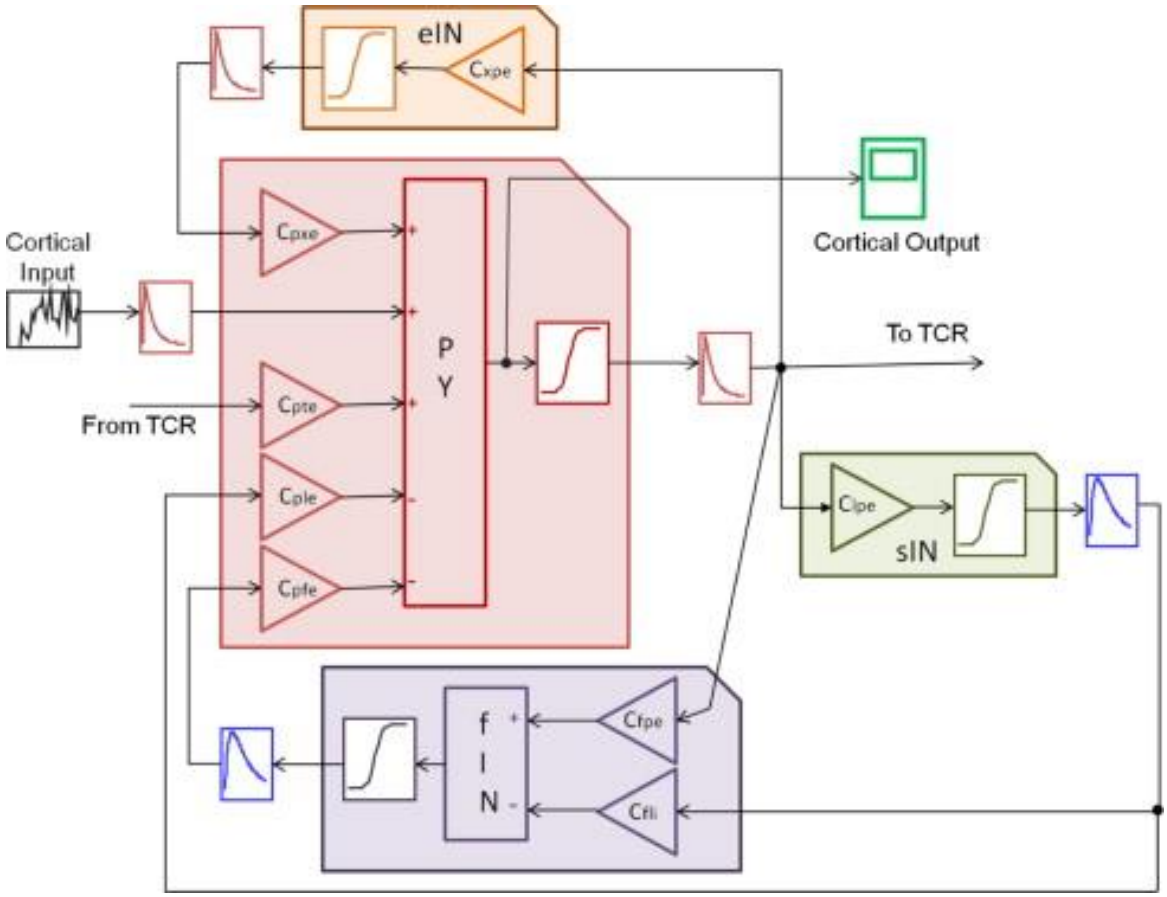

Figure 6.7 The computational model of the cortical module (Basabdatta Sen Bhattacharya, Coyle, and Maguire 2011b).

TABLE 6.3

BASAL VALUES FOR THE SYNAPTIC CONNECTIVITY PARAMETERS USED IN Eqs. 6.14-6.18 (Bhattacharya, Coyle, and Maguire 2011).

Module	Afferent (to)	Efferent (from)	Connectivity Parameter	Value
Cortical	PY	Cortical	C_{pce}	1
Cortical	PY	TCR	C_{pte}	80
Cortical	PY	fIN	C_{pfe}	108
Cortical	PY	eIN	C_{xpe}	135
Cortical	PY	sIN	C_{ple}	33.75
Cortical	fIN	PY	C_{fpe}	40.5
Cortical	fIN	sIN	C_{fli}	13.5
Cortical	sIN	PY	C_{lpe}	33.75
Cortical	eIN	PY	C_{Pxe}	108

6.5.2 Simulation and Signal Processing Methods for TCT Model

Since the TCT model is parameterised to simulate the real SSVEP signal, the output of the TCT model needs to be as close as possible to the real SSVEP recordings. The TCT model simulation is implemented using the 4th order Runge-Kutta ODE solver within the Simulink® environment in Matlab®. The total simulation time is 2.5 seconds at a sampling rate of 250 Hz. The output of the cortical module is considered as the TCT model's output. The obtained output vector is bandpass filtered with a Butterworth filter of order 10 with a lower and upper cut-off frequencies of 0.1 and 50 Hz, respectively. The power spectral density analysis (PSDA) defined in Eq. 6.6 is performed in Matlab® using a Welch periodogram, with a Hamming window of segment length consisting of 125 data points and overlap of 50%. For computational purpose, the output of the cortical module generated in Simulink® (simmodel) is passed through the PSDA (see Eq. 6.6). The outputs of the PSDA will then be used for computing the relative power RP using Eq. 6.7.

6.5.3 The Framework of Parameterising the TCT Model with Respect to the Real SSVEP Data

Our objective here is to perform a rigorous search for the suitable values of C_{xyz} in the TCT neural mass model such that the model output can be similar to the real recorded SSVEP signal. There are total of 19 synaptic connectivity parameters (C_{xyz}) in the TCT model. Through the proposed the parameterisation approach, we will be able to find a set of optimal values for the connectivity parameter set C_{xyz} for the TCT model such that the output of the TCT model is as close as possible to the real recorded SSVEP signal.

For brevity, in this section, the recorded SSVEP data from one of the subjects (please refer to Chapter 3 for more details on EEG data acquisition) is employed as the 'reference signal' without the loss of generality. Here, the 'reference signal' is defined as a signal to which the TCT model will be parameterised such that the simulated signal is as close as possible to the reference signal. As explained in Chapter 3, after recording a SSVEP data from a subject, the recorded data will go through steps including signal pre-processing, segmentation and feature extraction. Therefore, for each subject and a recorded signal, a dataset consisting of all features is represented as a matrix of size $S \times N$. S represents the number of segments and N represents the total of nine extracted features for each segment. These nine features are characterised by relative energy, wavelet entropy and the mean power of the detail coefficients, each of which is computed for three sub-bands, i.e. gamma, beta and alpha, which are all calculated based on Eqs. 3.6-3.11 in Chapter 3. Therefore, the reference signal is presented by each row in the matrix of size $S \times N$ and the TCT model will be parameterised to simulate a signal which is as similar as possible to the selected segment.

As the first attempt, in this project, only three features for each segment, including relative energies of gamma, beta and alpha bands, are investigated for parameterising the TCT model.

Moreover, we only include the SSVEP data corresponding to a visual stimulus at 10 Hz as the basal values of the connectivity parameter set C_{xyz} for the alpha band are known (Bhattacharya, Coyle, and Maguire 2011). Knowing the basal values will help in setting up the search space of C_{xyz}. The approach developed in this section can be applied to include more features, as well as to other bands provided more insight into the basal values of C_{xyz} in other frequency bands becomes available.

The parametrisation approach is based on a standard search strategy using a single objective GA. the connectivity parameter set C_{xyz} are encoded as real values. The search space of C_{xyz} is created using ±30% across the basal values. The only exception is made for C_{tre}, which is varied ±15% due to its relatively smaller value.

To parameterisingC_{xyz} of the TCT model, the objective function for the search algorithm will be formulated such that the TCT model's simulated signal would contain the same features (i.e. the three relative energies in different frequency bands) as those of the reference signal. Toward this, first, DWT will be applied to the cortical output of the TCT model to decompose the simulated signal and compute the wavelet coefficients. In implementation of DWT, the db4 filter is used and the decomposition level is set to 5, which are the same setting as the one used for decomposing the reference signal. In doing so, the simulated signal will be decomposed into five frequency bands, including noise (D_1), gamma (D_2), beta (D_3), alpha (D_4), and theta (D_5). The wavelet coefficients will then be computed accordingly (for more details please refer to Chapter 3).

The absolute distance of the relative energies of the simulated signal in alpha, gamma and beta frequency bands ($simE_{RengD_i}, i = 2, 3, 4$) from the corresponding relative energies of the reference signal ($refE_{RengD_i}, i = 2, 3, 4$) are computed using Eq. 6.19 and presented as, dif_{D_2}, dif_{D_3}, and dif_{D_4}, respectively.

In Eq. 6.19, $refE_{RengD_2}$, $refE_{RengD_3}$, $refE_{RengD_4}$, have already been computed in the feature extraction procedure for the recorded SSVEP signal in Chapter 3. $simE_{RengD_2}$, $simE_{RengD_3}$, $simE_{RengD_4}$ are computed using Eqs. 3.6–3.9 in Chapter 3

$$dif_{D_2} = \left|\left((simE_{RengD_2} - refE_{RengD_2}\right)\right|$$

$$dif_{D_3} = \left|\left((simE_{RengD_3} - refE_{RengD_3}\right)\right| \qquad (6.19)$$

$$dif_{D_4} = \left|\left((simE_{RengD_4} - refE_{RengD_4}\right)\right|$$

Finally, the objective function is defined as $objfit$ in Eq. 6.20 which is to minimise the summation of the respective absolute distances

$$objfit = (\sum_i^d dif_{D_i}, \;\; i = 2,3,4) \qquad (6.20)$$

The overall framework of parameterising the TCT model using the real SSVEP signal is presented in Figure 6.8.

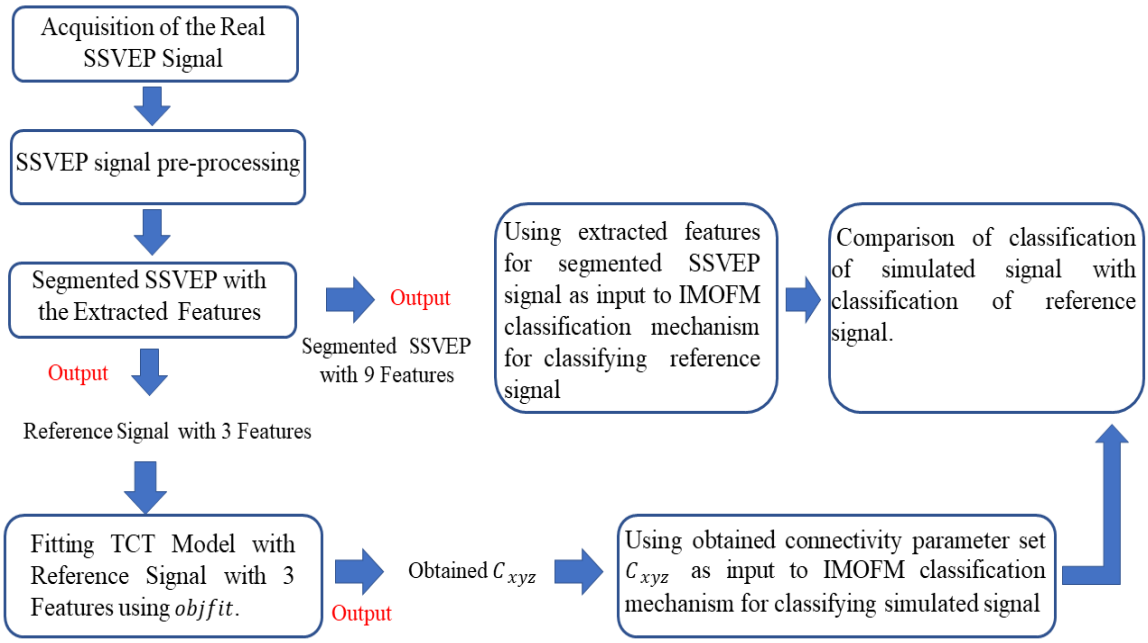

Figure 6.8 The overall framework of parameterising the TCT model.

In Figure 6.8, the search algorithm based on the GA will use the objective function defined in Eq. 6.20 to find the optimal connectivity parameter set C_{xyz} for the TCT model to simulate the selected reference signal.

6.5.4 Classification Using Labelled C_{xyz}

The obtained connectivity parameter set C_{xyz} together with their corresponding labels will form the new datasets to replace the original datasets that are formed based on the feature extraction method and reference signal as discussed in Chapter 3. These newly formed datasets will then be used as the training and testing sets for the proposed classification mechanism IMOFM in Chapter 4. The hypothesis made in this and the subsequent sections is that using biological plausible parameters such as C_{xyz} will increase the classification accuracy compared to those classifiers trained by the original datasets (see Chapter 4).

6.6 Computational Experiments for Parameterising TCT Model

In this section, experimental set-ups and results for parameterising the TCT model to simulate the real SSVEP signal are presented. Comparison of classification accuracy using the labelled C_{xyz} and original datasets are also provided.

Regarding the dataset used in the next sections, the dataset of CF1 for subject 2, are randomly partitioned into 75% training and 25% testing and considered as the feature dataset. Then apart from the label, each row in the training and testing of the feature dataset, will be replaced with the fitting C_{xyz} to form the training and the testing set of the labelled C_{xyz}.

In the classification experiment, to provide a rigorous analysis and comparison between the classification of feature dataset, and the labelled C_{xyz} dataset, training of each classifier with both dataset is separately run for 10 times. In each run, the training and the testing set of the

feature dataset, are randomised, and the randomised indices for training and the testing sets are stored. Then, for each run, the training and the testing labelled C_{xyz} datasets, are also randomised with the stored randomised indices.

6.6.1 Results on Parameterising the TCT Model and Classification Using Labelled C_{xyz}

The experiments for parameterising the TCT model are implemented using the GA toolbox in MATLAB. The experiments were carried out using the objective function defined in Eq. 6.20, with 50 generations and the population size of 50. As the TCT model also receives an external white noise as the input, the optimisation process is featured as a highly stochastic procedure. To address this problem, instead of using the robust optimisation scheme proposed in Section 6.3.3, we monitor the convergence of the mean fitness value in order to set up the empirical and required number of generations and population size. Other parameters are set as default. For this and the subsequent sections, since, the basal values for the synaptic connectivity parameters (TABLEs 6.1 and 6.3) for simulation of TCT model is based on the alpha band, therefore, the reference signals are chosen from the CF1 (CF1 covers the alpha band) of Subject 2 in Chapter 4.

Figure 6.9 illustrates comparison of the power spectral density of a reference signal to the simulated signal. Figure 6.9 (a) represents the comparison for Segment 51, and Figure 6.9 (b) shows the comparison for Segment 48. Please refer to Chapter 3 for segmentation of the recorded EEG signals.

As shown in Figure 6.9, for both segments the simulated signal has captured the main characteristics of the reference signals which is a significant peak around 10 Hz frequency.

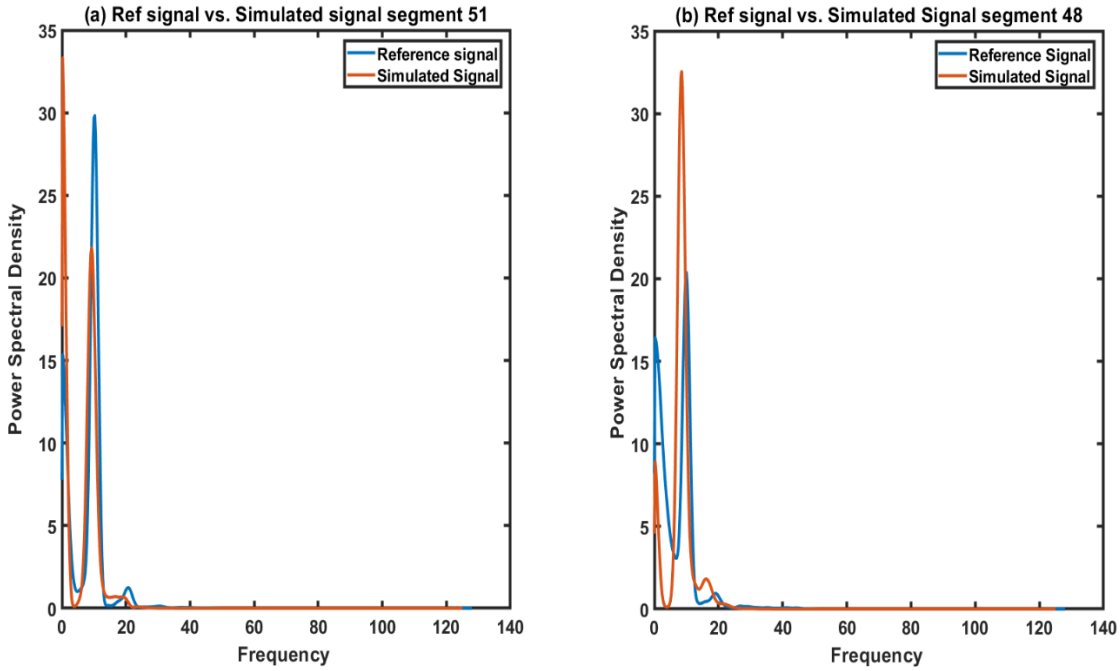

Figure 6.9 The comparison of the power spectral density of the reference signals against the simulated signals. (a) Segment 51, and (b) Segment 48.

IMOFM-C classification mechanism is employed for classification of both the feature dataset and labelled C_{xyz} dataset. For evaluation purpose, all algorithms in comparison have been run 10 times and the results are averaged. Since the IMOFM-C classification mechanism is based on the multi-objective fuzzy approach, it evolves a Pareto front of solutions at each run. Therefore, two evaluation procedures mentioned in Section 5.4.2 in Chapter 5 are carried out to evaluate the performances of training and testing based on the labelled C_{xyz} dataset. The results are shown in TABLE 6.4.

As shown in TABLE 6.4, the classification performances based on the labelled C_{xyz} dataset in four classification performance measures for both training and testing performs poorly comparing to the classification performances based on the feature dataset. However, the results still demonstrate the potential of using labelled C_{xyz} for classification as ~70% accuracy is achieved in testing.

TABLE 6.4

COMPARSION OF CLASSIFICATION OF THE FEATURE DATASET WITH CLASSIFICATION OF LABELLED C_{xyz} FOR THE BEST CLASSIFIER AND THE ENSEMBLE CLASSIFICATION ON TEN PARTITIONS OF THE CF1 CLASSIFIER DATASET FOR SUBJECT 2.

Classifiers IMOFM-C	**Accuracy (TR/TS)**	**Sensitivity (TR/TS)**	**Specificity (TR/TS)**	**Miss-points (TR/TS)**
Best feature set	1/0.9473	1/0.9054	1/0.9054	0/2.90
Ensemble feature set	1/0.9491	1/0.9067	1/0.9067	0/2.80
Best labelled C_{xyz}	0.9963/0.6945	0.9932/0.5921	0.9932/0.5921	0.6/18.80
Ensemble labelled C_{xyz}	0.9933/0.7073	0.9875/0.6008	0.9875/0.6008	1.10/16.10

Potential reasons for inferior performances may include: (1) The proposed fitting framework in Section 6.5.3 needs to include other frequency domain features such as peaks amplitudes and positions, and the area under the power spectrum curve; (2) the TCT model needs to be further developed for generation of SSVEP data.

6.7 Conclusion

In this chapter, first, a robust optimisation approach is introduced for parameterising a thalamic neural mass model that simulates brain oscillations such as those observed in EEG and local field potentials. Second, preliminary experiments to evaluate the feasibility of the proposed framework for parameterising the TCT neural mass model with respect to the real SSVEP data were carried out.

Chapter 7

Conclusions and Future Work

Every scientific endeavour tries to find the answers to the problems at hand and in doing so, raises many others. The work presented in this thesis is not an exception. It tries to answer the following three questions:

1. How to improve transparency and generalisation capability of FRBS-based classifier through metaheuristics in order to address multiclass SSVEP EEG data classification problems.

2. How to improve further the classification performance of FRBS-based binary classifiers though preference-based metaheuristics and ensemble classification mechanism,

3. How to bridge the gap between BCI and NMMs.

This final chapter summarises what has been achieved in answering the above three questions and what are the open questions that deserve further research efforts.

7.1 Conclusions

To answer the first question, an Immune Inspired Multi-Objective Fuzzy Modelling (IMOFM) mechanism, which was originally designed for regression problems, is adapted for classification of multiclass SSVEP EEG data. IMOFM features a multi-stage modelling procedure combining a gradient based local search and a metaheuristic based multi-objective search algorithm. Due to this multi-stage modelling procedure, IMOFM leads to not only a high prediction

accuracy but also a simplified model structure. The main aspects that have been investigated in the adapted IMOFM Classification (IMOFM-C) mechanism are as follow:

1. The multi-objective modelling framework of IMOFM has been exploited to elicit a set of diverse and accurate FRBSs which are later on used in designing effective ensemble classifiers.

2. Three different combinations of objective functions, including 'accuracy vs. diversity (PFC)', 'accuracy vs. diversity (NCL)' and 'accuracy vs. interpretability' were investigated in order to get the most diversified set of classifiers. The results show that implementing the IMOFM-C mechanism with 'accuracy vs. interpretability' as the objective functions has not only encouraged the diversity of the solutions but also evolved significantly more accurate FRBSs comparing to the other two combinations.

3. A rigorous parameter analysis has been carried out and the main parameters investigated were included: a) the sufficient number of generations to warrant a convergence of the IMOFM-C mechanism; and b) the predefined maximum and minimum number of rules for FRBSs in IMOFM-C.

4. One-Against-One (OAO) decomposition-based approach is used to decompose the SSVEP EEG multi-class problem in to six binary classification problems. To aggregate the binary classifiers, the Decision Directed Acyclic Graph (DDAG)-Distance aggregation approach is proposed.

   The developed IMOFM-C mechanism have been applied to two benchmark datasets including, Australian Credit Card dataset and Diabetes dataset as well as the SSVEP EEG data recorded from 4 subjects (2 males and 2 females). The performance of IMOFM-C has been compared to those of various baseline classification approaches, such as ANFIS,

ANNs, CART decision tree, and SVM. The results of the comparison show, the proposed IMOFM-C based multi-class classification mechanism presents superior performance in terms of both accuracy and interpretability.

In order to answer the second question, a preference-based ensemble classification framework known as IMOFM-CP is developed based on the IMOFM-C mechanism. In the proposed IMOFM-CP, the PAIA algorithm (see Chapter 4) is modified to incorporate the preference angle and reference information-based dominance (ar-dominance) (Yi et al. 2018).In designing the IMOFM-CP mechanism the following points are considered and investigated:

1. The modified PAIA algorithm increases the convergence speed of the population and reduces the number of solutions in the non-preferred region (Yi et al. 2018). Therefore, classifiers in the non-preferred region, i.e. the region that contains less accurate classification models, are automatically filtered out. On the contrary, classifiers in the preferred region, i.e. the region that contains accurate enough classifiers, will undergo a more focused search, further improving their classification accuracy.

2. IMOFM-CP is designed to promote the diversity among the elicited classification models and fully utilise the power of each individual classification model as a member of a committee classifier (i.e. ensemble classifier). Aggregating a set of elicited classifiers leads to more accurate classification performance than that of the individual classifier or, in the worst cases, equally good classification performance. In this work our proposed DDAG-Distance aggregation approach is applied to work with IMOFM-CP, since it has shown to have a better performance comparing to the conventional aggregation approaches such as, voting strategy, Averaged voting strategy, and Decision Directed Acyclic Graph (DDAG).

The developed IMOFM-CP mechanism have been applied to the same benchmark datasets, as well as the SSVEP EEG datasets. The performance of IMOFM-CP has been compared to IMOFM-C and the same baseline classification approaches. The proposed IMOFM-CP presents superior performance compared to that of IMOFM-C.

Finally, in order to answer the third question, first, the gap existing between BCI and NMMs has been identified. Traditional ways of studying BCI often include acquiring appropriate brain signals and establishing classification models based on the acquired data (Mason and Birch 2003). Although the acquired signal types and technologies for classification are varied, they represent state-of-the-art in BCI towards interfacing human intentions with the outside world. However, the procedure mentioned above suffers from the following limitations:

1. Different BCI applications often require different brain signals which are categorised in five frequency bands, namely a) Delta (<4 HZ), b) Theta (4-7 HZ), c) Alpha (8-15 HZ), d) Beta (16-31 HZ), and e) Gamma (32+ HZ) (Buzsáki 2006).

2. A large number of experimental data has to be acquired in order to achieve satisfactory classification accuracy.

Both of the above limitations put a significant constraint on studying BCI as in practice they inevitably lead to a fairly large number of participating subjects and experiments. Therefore, using neural mass models can help to overcome these limitations and, the idea of combining these two close areas (BCI and NMMs) would bridge the identified gap and provide more support to design a more reliable BCI system.

NMMs can help to model the region of the interest (ROI) of the human brains. Furthermore, they allow us to reconstruct ROIs that are inaccessible to measurement. This de facto offers an effective way of simulating various brain signals across different subjects under varying testing

environments (e.g. with different noise and interruptions). In order to make a NMM to simulate the specific brain signals, part or all of the model's parameters need to be parameterised. Towards this, a robust optimisation approach is introduced for parameterising the thalamic module as a part of the thalamo-cortico-thalamic (TCT) circuitry based NMM NMM (Basabdatta Sen Bhattacharya, Coyle, and Maguire 2011b).

1. The proposed robust parameterisation approach combines a robust single-objective genetic algorithm and a clustering algorithm. A set of objective and penalty functions are devised to overcome the extrinsic uncertainty in inputs.

2. he clustering algorithm facilitates the identification of robust regions such that small variations can be accommodated. Preliminary results suggest that multiple robust synaptic connectivity parameter regions exist, which are distinct from the suggested basal values in literature.

3. The direct implication of 2. is for more accurate predication of brain-disorder diseases, as well as to investigate if those robust regions are consistent across different subjects in a varying environment and can be used to design a more reliable human-machine interface.

7.2 Future Research Directions

As mentioned at the beginning of this chapter, when the immediate problems are solved, new problems will arise and they, too, should be solved. Such problems remain as open questions and are discussed in the following part.

- **Increasing further the diversity of classifiers generated by the IMOFM-CP mechanism:** It is reckoned that the diversity of the classifiers can be further improved by fully integrating diversity measures, such as PFC or NCL, into IMOFM-CP. Currently, PFC and

NCL have been used in IMOFM-C as one of the objectives. The selection procedure in IMOFM-C/CP remains the same as that of IMOFM which was designed to encourage simplification of FRBS structure and is directly linked to the interpretability objective. However, diversity comes into play not only via structure changes but also through parameter tuning. Simplifying the structure naturally leads to a reduced number of parameters, resulting in a less degree of freedom in achieving overall diversity. Therefore, it is envisioned that in future corresponding structure preserving mechanisms should also be added to the selection procedure so that FRBSs with similar structures can be encouraged to search different regions through parameter tuning. This way the classifiers will go through another round of diversification and hence a higher level of diversity for the classifiers can be achieved.

- **Using NMMs for the further improvement of classification accuracy:** once NMMs are parameterised, they can be used to generate additional EEG signals. Therefore, the classifiers based on IMOFM-CP can be trained using two types of data sets: a) simulated data using the parameterised NMM; and b) real EEG data recorded from the subjects during the experimental procedures. Furthermore, the classifiers will be trained not only with the data sets belonging to various subjects but also with the data set reflecting varying environment and subjects' conditions by varying the parameters within NMMs. Therefore, a potentially significant improvement in terms of classification accuracy under various conditions is expected with the incorporation of the NMMs.

- **Developing on-line NMMs fitting procedure for classification purpose:** The robust regions identified through the parameterisation procedure can also be used as a direct reference for classification purpose. In that case, an analytical solution to parameterising NMMs (Hashemi 2016) can be developed to fit NMMs to the on-line EEG recordings. The on-line identified parameters of NMMs will then be compared with the off-line identified robust

regions through e.g. the nearest neighbour search approach. Since the off-line robust regions take varying conditions in subjects and environment into account and solving an analytic solution is computationally efficient, one could achieve high cross-subject on-line classification performance.

- **Investigating the biological meanings associated with the identified parameters through parameterisation of NMMs:** Neural mass models, are considered as an appropriate approach to acquire a deep insight into high-level brain rhythms, resulting in a better simulation of brain dynamics. Therefore, studying how the identified robust regions can match the biological definition of brain rhythms will help us to make a NMMs a more reliable framework of reproducing real EEG data.

Bibliography

Abdulkader, Sarah N., Ayman Atia, and Mostafa-Sami M. Mostafa. 2015. “Brain Computer Interfacing: Applications and Challenges.” Egyptian Informatics Journal 16 (2): 213–30. https:

Acharya, Jayant N., Abeer Hani, Janna Cheek, Partha Thirumala, and Tammy N. Tsuchida. 2016. “American Clinical Neurophysiology Society Guideline 2.” *Journal of Clinical Neurophysiology* 33 (4): 308–11.

Adeli, Hojjat, Ziqin Zhou, and Nahid Dadmehr. 2003. “Analysis of EEG Records in an Epileptic Patient Using Wavelet Transform.” *Journal of Neuroscience Methods* 123 (1): 69–87.

Akilandeswari, K., and G. M. Nasira. 2014. “Bagging of EEG Signals for Brain Computer Interface.” In *2014 World Congress on Computing and Communication Technologies*, 71–75. IEEE.

Al-Fahoum, Amjed S, and Ausilah A Al-Fraihat. 2014. “Methods of EEG Signal Features Extraction Using Linear Analysis in Frequency and Time-Frequency Domains.” *ISRN Neuroscience* 2014 .

Al-Jumeily, Dhiya, Shamaila Iram, Francois-Benois Vialatte, Paul Fergus, and Abir Hussain. 2015. “A Novel Method of Early Diagnosis of Alzheimer’s Disease Based on EEG Signals.” *TheScientificWorldJournal* 2015.

Amiri, Setare, Ahmed Rabbi, Leila Azinfar, and Reza Fazel-Rezai. 2013. “A Review of P300, SSVEP, and Hybrid P300/SSVEP Brain- Computer Interface Systems.” In *Brain-Computer Interface Systems - Recent Progress and Future Prospects*.

Anand, Ashish, and P.N. Suganthan. 2009. “Multiclass Cancer Classification by Support Vector Machines with Class-Wise Optimized Genes and Probability Estimates.” *Journal of Theoretical Biology* 259 (3): 533–40.

Aran, Oya, and Lale Akarun. 2010. “A Multi-Class Classification Strategy for Fisher Scores: Application to Signer Independent Sign Language Recognition.” *Pattern Recognition* 43 (5)

Arasaratnam, I., and S. Haykin. 2009. “Cubature Kalman Filters.” *IEEE Transactions on Automatic Control* 54 (6): 1254–69.

Asch, Vincent Van. 2013. “Macro-and Micro-Averaged Evaluation Measures [[BASIC DRAFT]].”

Benítez, J. M., J. L. Castro, and I. Requena. 1997. “Are Artificial Neural Networks Black Boxes?” *IEEE Transactions on Neural Networks* 8 (5): 1156–64.

Bhattacharya, Basab, Damien Coyle, and LP Maguire. 2012. “Assessing Alpha Band Event-Related Synchronisation/Desynchronisation Using a Bio-Inspired Computational Model.” *Journal of Universal Computer Science*.

Bhattacharya, Basabdatta sen, Damien Coyle, and Liam Maguire. 2011. “Synaptic Deficiency in Alzheimer Disease and Its Implications on Thalamic and Cortical Oscillations: A Study Using a Neural Mass Model.”

Bhattacharya, Basabdatta Sen, Damien Coyle, and Liam P. Maguire. 2011a. “Assessing Retino-Geniculo-Cortical Connectivities in Alzheimer’s Disease with a Neural Mass Model.” *IEEE SSCI 2011 - Symposium Series on Computational Intelligence - CCMB 2011: 2011 IEEE Symposium on Computational Intelligence, Cognitive Algorithms, Mind, and Brain*, 159–63.

Bhattacharya, Basabdatta Sen, Damien Coyle, and Liam P Maguire. 2011b. “A Thalamo-Cortico-Thalamic Neural Mass Model to Study Alpha Rhythms in Alzheimer’s Disease.” *Neural Networks : The Official Journal of the International Neural Network Society* 24 (6): 631–45.

Bhowan, Urvesh, Mark Johnston, Mengjie Zhang, and Xin Yao. 2013. “Evolving Diverse Ensembles Using Genetic Programming for Classification With Unbalanced Data.” *IEEE Transactions on Evolutionary Computation* 17 (3): 368–86.

Brainard, D H. 1997. “The Psychophysics Toolbox.” *Spatial Vision* 10 (4): 433–36.

Breiman, Leo. 1996. “Bagging Predictors.” *Machine Learning* 24 (2): 123–40.

Brown, Gavin, and Gavin Brown. 2004. “Diversity in Neural Network Ensembles.”

Buzsáki, György. 2006. *Rhythms of the Brain*. Oxford University Press.

Carpenter, Gail A., and Stephen Grossberg. 1991. *Pattern Recognition by Self-Organizing*

Neural Networks. MIT Press.

Chandra, Arjun, and Xin Yao. 2004. "DIVACE: Diverse and Accurate Ensemble Learning Algorithm." In , 619–25. Springer, Berlin, Heidelberg.

Chandra, Arjun, and Xin Yao. 2006. "Ensemble Learning Using Multi-Objective Evolutionary Algorithms." *Journal of Mathematical Modelling and Algorithms* 5 (4): 417–45.

Chen, Jun, Michael Gallimore, Chris Bingham, Mahdi Mahfouf, and Yu Zhang. 2014. "Intelligent Data Compression, Diagnostics and Prognostics Using an Evolutionary-Based Clustering Algorithm for Industrial Machines." NOVA Science Publisher.

Chen, Jun, and Mahdi Mahfouf. 2006. "A Population Adaptive Based Immune Algorithm for Solving Multi-Objective Optimization Problems." In , 280–93. Springer, Berlin, Heidelberg.

Chen, Jun, and Mahdi Mahfouf. 2012. "Improving Transparency in Approximate Fuzzy Modeling Using Multi-Objective Immune-Inspired Optimisation." *International Journal of Computational Intelligence Systems* 5 (2): 322–42.

Chen, Xiaogang, Yijun Wang, Masaki Nakanishi, Xiaorong Gao, Tzyy-Ping Jung, and Shangkai Gao. 2015. "High-Speed Spelling with a Noninvasive Brain-Computer Interface." *Proceedings of the National Academy of Sciences of the United States of America* 112 (44): E6058-67.

Cheng, Ming, Xiaorong Gao, Shangkai Gao, and Dingfeng Xu. 2002. "Design and Implementation of a Brain-Computer Interface with High Transfer Rates." *IEEE Transactions on Biomedical Engineering* 49 (10): 1181–86.

Cherkassky, V. 2005. "A Combined SVM and LDA Approach for Classification." In *Proceedings. 2005 IEEE International Joint Conference on Neural Networks, 2005.*, 3:1455–59. IEEE.

Chiu, Stephen L. 1994. "Fuzzy Model Identification Based on Cluster Estimation." *Journal of Intelligent & Fuzzy Systems: Applications in Engineering and Technology* 2 (3): 267–78.

Comerchero, M D, and J Polich. 1999. "P3a and P3b from Typical Auditory and Visual Stimuli." *Clinical Neurophysiology : Official Journal of the International Federation of Clinical Neurophysiology* 110 (1): 24–30.

Cona, F, M Zavaglia, M Massimini, M Rosanova, and M Ursino. 2011. “A Neural Mass Model of Interconnected Regions Simulates Rhythm Propagation Observed via TMS-EEG.” *NeuroImage* 57 (3): 1045–58.

Cona, Filippo, Melissa Zavaglia, Laura Astolfi, Fabio Babiloni, and Mauro Ursino. 2009. “Changes in Eeg Power Spectral Density and Cortical Connectivity in Healthy and Tetraplegic Patients during a Motor Imagery Task.” *Computational Intelligence and Neuroscience* 2009.

Cordón, Oscar, María José del Jesus, and Francisco Herrera. 1999. “A Proposal on Reasoning Methods in Fuzzy Rule-Based Classification Systems.” *International Journal of Approximate Reasoning* 20 (1): 21–45.

Coyle, Damien, Basabdatta S. Bhattacharya, Xin Zou, KongFatt Wong-Lin, Kamal Abuhassan, and Liam Maguire. 2014. “Neural Circuit Models and Neuropathological Oscillations.” In *Springer Handbook of Bio-/Neuroinformatics*, 673–702. Berlin, Heidelberg: Springer Berlin Heidelberg.

Cvetkovic, Dean, Elif Derya Übeyli, and Irena Cosic. 2008. “Wavelet Transform Feature Extraction from Human PPG, ECG, and EEG Signal Responses to ELF PEMF Exposures: A Pilot Study.” *Digital Signal Processing* 18 (5): 861–74.

Deb, K. 2001. *Multi-Objective Optimization Using Evolutionary Algorithms | Wiley*. Wiley.

Deb, K., A. Pratap, S. Agarwal, and T. Meyarivan. 2002. “A Fast and Elitist Multiobjective Genetic Algorithm: NSGA-II.” *IEEE Transactions on Evolutionary Computation* 6 (2): 182–97.

Deb, Kalyanmoy, and J. Sundar. 2006. “Reference Point Based Multi-Objective Optimization Using Evolutionary Algorithms.” In *Proceedings of the 8th Annual Conference on Genetic and Evolutionary Computation - GECCO ’06*, 635. New York, New York, USA: ACM Press.

Deco, Gustavo, Viktor K Jirsa, Peter A Robinson, Michael Breakspear, and Karl Friston. 2008. “The Dynamic Brain: From Spiking Neurons to Neural Masses and Cortical Fields.” *PLoS Computational Biology* 4 (8): e1000092.

Delorme, Arnaud, and Scott Makeig. 2004. “EEGLAB: An Open Source Toolbox for Analysis of Single-Trial EEG Dynamics Including Independent Component Analysis.” *Journal of Neuroscience Methods* 134 (1): 9–21.

Deng, Houtao, George Runger, Eugene Tuv, and Martyanov Vladimir. 2013. “A Time Series Forest for Classification and Feature Extraction.” *Information Sciences* 239 (February): 142–53.

Dietterichl, Thomas G. 2002. “Ensemble Learning.”

Eichelberger, Robert Kyle, and Victor S. Sheng. 2013. “Does One-Against-All or One-Against-One Improve the Performance of Multiclass Classifications?” *AAAI*.

Einicke, G.A., and L.B. White. 1999. “Robust Extended Kalman Filtering.” *IEEE Transactions on Signal Processing* 47 (9): 2596–99.

F.H.Lopes Da Silva, Ernest Niedrmeyer. 1982. “Electroencephalography: Basic Principles, Clinical Applications and Related Fields: Amazon.Co.Uk: Ernst Niedermeyer, F.H.Lopes Da Silva: 9780781751261: Books.” 1982.

Fabien, Lotte, Lécuyer Anatole, Lamarche Fabrice, and Arnaldi Bruno. 2007. “Studying the Use of Fuzzy Inference Systems for Motor Imagery Classification.” *IEEE Transactions on Neural Systems and Rehabilitation Engineering : A Publication of the IEEE Engineering in Medicine and Biology Society* 15 (2): 322–24.

Fayyad, Usama, Gregory Piatetsky-Shapiro, and Padhraic Smyth. 1996. “From Data Mining to Knowledge Discovery in Databases.” *AI Magazine*.

Fernández-Delgado, Manuel, Eva Cernadas, Senén Barro, Dinani Amorim, and Amorim Fernández-Delgado. 2014. “Do We Need Hundreds of Classifiers to Solve Real World Classification Problems?” *Journal of Machine Learning Research*. Vol. 15.

Forslund, Pontus. 2003. “A Neural Network Based Brain-Computer Interface for Classification of Movement Related EEG.”

Freund, Yoav, and Robert E Schapire. 1997. “A Decision-Theoretic Generalization of On-Line Learning and an Application to Boosting.” *Journal of Computer and System Sciences* 55 (1): 119–39.

Galar, Mikel, Alberto Fernández, Edurne Barrenechea, Humberto Bustince, and Francisco Herrera. 2011. “An Overview of Ensemble Methods for Binary Classifiers in Multi-Class Problems: Experimental Study on One-vs-One and One-vs-All Schemes.” *Pattern Recognition* 44 (8): 1761–76.

Gandhi, Vaibhav. 2014. *Brain-Computer Interfacing for Assistive Robotics: Electroencephalograms, Recurrent Quantum Neural Networks, and User-Centric Graphical Interfaces*. Elsevier Science.

Garg, Girisha, Vijander Singh, J R P Gupta, and A P Mittal. 2011. "RELATIVE WAVELET ENERGY AS A NEW FEATURE EXTRACTOR FOR SLEEP CLASSIFICATION USING EEG SIGNALS." *International Journal of Biomedical Signal Processing* 2 (2): 75–80.

Georgopoulos, A P, J F Kalaska, R Caminiti, and J T Massey. 1982. "On the Relations between the Direction of Two-Dimensional Arm Movements and Cell Discharge in Primate Motor Cortex." *The Journal of Neuroscience : The Official Journal of the Society for Neuroscience* 2 (11): 1527–37.

Gert Pfurtscheller, Brendan Allison, Bernard Garimann. 2010. "Brain–Computer Interfaces: A Gentle Introduction." 2010.

Goldberg, D E. 1989. *Genetic Algorithms in Search, Optimization, and Machine Learning*. *Addison Wesley*.

Gratton, G, M G Coles, and E Donchin. 1983. "A New Method for Off-Line Removal of Ocular Artifact." *Electroencephalography and Clinical Neurophysiology* 55 (4): 468–84.

Grimbert, François, and Olivier Faugeras. 2006. "Bifurcation Analysis of Jansen's Neural Mass Model." *Neural Computation* 18 (12): 3052–68.

Guan, Cuntai, Manoj Thulasidas, and Jiankang Wu. 2004. "High Performance P300 Speller for Brain-Computer Interface." In *2004 IEEE International Workshop on Biomedical Circuits and Systems*.

Guler, I., and E.D. Ubeyli. 2007. "Multiclass Support Vector Machines for EEG-Signals Classification." *IEEE Transactions on Information Technology in Biomedicine* 11 (2): 117–26.

Hadka, David, and Patrick Reed. 2013. "Borg: An Auto-Adaptive Many-Objective Evolutionary Computing Framework." *Evolutionary Computation* 21 (2): 231–59.

Hashemi, Meysam. 2016. "Analytical and Numerical Studies of Thalamo-Cortical Neural Population during General Anesthesia," January.

Hebbink, G.J. 2014. "Activity Types in a Neural Mass Model."

Hofmann, Martin. 2006. “Support Vector Machines-Kernels and the Kernel Trick An Elaboration for the Hauptseminar ‘Reading Club: Support Vector Machines.’”

Holland, John H. 1992. “Genetic Algorithms.” *Scientific American*.

Honeycutt, Rebecca L. 1992. “Stochastic Runge-Kutta Algorithms. I. White Noise.” *Physical Review A* 45 (2): 600–603.

Horn, S C Van, A Erişir, and S M Sherman. 2000. “Relative Distribution of Synapses in the A-Laminae of the Lateral Geniculate Nucleus of the Cat.” *The Journal of Comparative Neurology* 416 (4): 509–20.

Htike, Kyaw Kyaw, and Othman O. Khalifa. 2010. “Comparison of Supervised and Unsupervised Learning Classifiers for Human Posture Recognition.” In *International Conference on Computer and Communication Engineering (ICCCE’10)*, 1–6. IEEE.

Ianez, Eduardo, Jose M. Azorin, Andres Ubeda, Eduardo Fernandez, and Jose L. Sirvent. 2010. “LDA-Based Classifiers for a Mental Tasks-Based Brain-Computer Interface.” In *2010 IEEE International Conference on Systems, Man and Cybernetics*, 546–51. IEEE.

Ishibuchi, Hisao, Hiroyuki Masuda, Yuki Tanigaki, and Yusuke Nojima. 2014. “Difficulties in Specifying Reference Points to Calculate the Inverted Generational Distance for Many-Objective Optimization Problems.” In *2014 IEEE Symposium on Computational Intelligence in Multi-Criteria Decision-Making (MCDM)*, 170–77. IEEE.

Ishibuchi, Hisao, Tomoharu Nakashima, and Tadahiko Murata. 2001. “Three-Objective Genetics-Based Machine Learning for Linguistic Rule Extraction.” *Information Sciences* 136 (1–4): 109–33.

Ishibuchi, Hisao, and Yusuke Nojima. 2006. “Fuzzy Ensemble Design through Multi-Objective Fuzzy Rule Selection.” In *Multi-Objective Machine Learning*, 507–30. Berlin, Heidelberg: Springer Berlin Heidelberg.

Ishibuchi, Hisao, and Takashi Yamamoto. 2004. “Fuzzy Rule Selection by Multi-Objective Genetic Local Search Algorithms and Rule Evaluation Measures in Data Mining.” In *Fuzzy Sets and Systems*, 141:59–88.

Jahankhani, Pari, Vassilis Kodogiannis, and Kenneth Revett. 2006. “EEG Signal Classification Using Wavelet Feature Extraction and Neural Networks.” In *IEEE John Vincent Atanasoff*

2006 International Symposium on Modern Computing (JVA'06), 120–24. IEEE.

Jain, Anil K. 2010. "Data Clustering: 50 Years beyond K-Means." *Pattern Recognition Letters* 31 (8): 651–66.

Jang, Jyh Shing Roger. 1993. "ANFIS: Adaptive-Network-Based Fuzzy Inference System." *IEEE Transactions on Systems, Man and Cybernetics* 23 (3): 665–85.

Jansen, B H, and V G Rit. 1995. "Electroencephalogram and Visual Evoked Potential Generation in a Mathematical Model of Coupled Cortical Columns." *Biological Cybernetics* 73 (4): 357–66.

Jansen, Ben H., George Zouridakis, and Michael E. Brandt. 1993. "A Neurophysiologically-Based Mathematical Model of Flash Visual Evoked Potentials." *Biological Cybernetics* 68 (3): 275–83.

Japkowicz, Nathalie, and Shaju Stephen. 2002. "The Class Imbalance Problem: A Systematic Study." *Intelligent Data Analysis* 6 (5): 429–49.

Jin, Y. 2000. "Fuzzy Modeling of High-Dimensional Systems: Complexity Reduction and Interpretability Improvement." *IEEE Transactions on Fuzzy Systems* 8 (2): 212–21.

Jin, Y, and B Sendhoff. 2002. "Incorporation Of Fuzzy Preferences Into Evolutionary Multiobjective Optimization."

Jin, Yaochu. 2000. "Fuzzy Modeling of High-Dimensional Systems: Complexity Reduction and Interpretability Improvement." *IEEE Transactions on Fuzzy Systems* 8 (2): 212–21.

Julier, Simon J., and Jeffrey K. Uhlmann. 1997. "<title>New Extension of the Kalman Filter to Nonlinear Systems</Title>." In *AeroSense '97*, edited by Ivan Kadar, 182–93. International Society for Optics and Photonics.

Jun, Chen. 2010. "Biologically Inspired Optimisation Algorithms for Transparent Knowledge Extraction Allied to Engineering Materials Processing," January.

Kalman, R. E. 1960. "A New Approach to Linear Filtering and Prediction Problems." *Journal of Basic Engineering* 82 (1): 35.

Kandel, Eric R. 2000. "Principles of Neural Science, Fifth Edition (Principles of Neural Science (Kandel)): 9780071390118: Medicine & Health Science Books" .

Kaufman, L., and Peter Rousseeuw. 1987. “Clustering by Means of Medoids,” January.

Kelly, Simon P, Edmund C Lalor, Richard B Reilly, and John J Foxe. 2005. “Visual Spatial Attention Tracking Using High-Density SSVEP Data for Independent Brain-Computer Communication.” *IEEE Transactions on Neural Systems and Rehabilitation Engineering : A Publication of the IEEE Engineering in Medicine and Biology Society* 13 (2): 172–78.

Kleiner, M, D Brainard, Denis Pelli, A Ingling, R Murray, and C Broussard. 2007. *Perception. Perception*. Vol. 36. [Pion Ltd.].

Knowles, Joshua, and David Corne. 1999. “The Pareto Archived Evolution Strategy: A New Baseline Algorithm for Pareto Multiobjective Optimisation.” In *Proceedings of the 1999 Congress on Evolutionary Computation, CEC 1999*, 1:98–105. IEEE Computer Society.

Köppen, Mario, and Kaori Yoshida. 2007. “Substitute Distance Assignments in NSGA-II for Handling Many-Objective Optimization Problems.” In *Evolutionary Multi-Criterion Optimization*, 727–41. Berlin, Heidelberg: Springer Berlin Heidelberg.

Kuncheva, Ludmila I., and Christopher J. Whitaker. 2003. “Measures of Diversity in Classifier Ensembles and Their Relationship with the Ensemble Accuracy.” *Machine Learning* 51 (2): 181–207.

Lemm, Steven, Klaus Robert Müller, and Gabriel Curio. 2009. “A Generalized Framework for Quantifying the Dynamics of EEG Event-Related Desynchronization.” *PLoS Computational Biology* 5 (8): e1000453. https://doi.org/10.1371/journal.pcbi.1000453.

Leow, R. S., F. Ibrahim, and M. Moghavvemi. 2007. “Development of a Steady State Visual Evoked Potential (SSVEP)-Based Brain Computer Interface (BCI) System.” In *2007 International Conference on Intelligent and Advanced Systems*, 321–24. IEEE.

Liljenström, Hans. 2012. “Mesoscopic Brain Dynamics.” *Scholarpedia* 7 (9): 4601.

Lin, Zhonglin, Changshui Zhang, Wei Wu, and Xiaorong Gao. 2007. “Frequency Recognition Based on Canonical Correlation Analysis for SSVEP-Based BCIs.” *IEEE Transactions on Biomedical Engineering* 54 (6): 1172–76.

Liu, Xian, and Qing Gao. 2013. “Parameter Estimation and Control for a Neural Mass Model Based on the Unscented Kalman Filter.” *Physical Review. E, Statistical, Nonlinear, and Soft Matter Physics* 88 (4): 042905.

Liu, Xian, Hui-Jun Liu, Ying-Gan Tang, Qing Gao, and Zhan-Ming Chen. 2014. “Fuzzy Adaptive Unscented Kalman Filter Control of Epileptiform Spikes in a Class of Neural Mass Models.” *Nonlinear Dynamics* 76 (2): 1291–99.

Liu, Y., and X. Yao. 1999. “Ensemble Learning via Negative Correlation.” *Neural Networks* 12 (10): 1399–1404.

Lopes da Silva, F. H., A. Hoeks, H. Smits, and L. H. Zetterberg. 1974. “Model of Brain Rhythmic Activity.” *Kybernetik* 15 (1): 27–37.

Lopes da Silva, F H, A van Rotterdam, P Barts, E van Heusden, and W Burr. 1976. “Models of Neuronal Populations: The Basic Mechanisms of Rhythmicity.” *Progress in Brain Research* 45 (January): 281–308.

López-Cuevas, Armando, Bernardino Castillo-Toledo, Laura Medina-Ceja, and Consuelo Ventura-Mejía. 2015. “State and Parameter Estimation of a Neural Mass Model from Electrophysiological Signals during the Status Epilepticus.” *NeuroImage* 113 (June): 374–86.

Lorincz, Magor L, Katalin A Kékesi, Gábor Juhász, Vincenzo Crunelli, and Stuart W Hughes. 2014. “Lorincz et Al 2009.”

Lotte, F, M Congedo, A Lécuyer, F Lamarche, and B Arnaldi. 2007. “A Review of Classification Algorithms for EEG-Based Brain-Computer Interfaces.” *Journal of Neural Engineering* 4 (2): R1–13.

Luijtelaar, L. Huang and G. van. 2013. *Brain-Computer Interface Systems - Recent Progress and Future Prospects*. Edited by Reza Fazel-Rezai. InTech.

Mallat, S.G. 1989. “A Theory for Multiresolution Signal Decomposition: The Wavelet Representation.” *IEEE Transactions on Pattern Analysis and Machine Intelligence* 11 (7): 674–93.

Malmivuo, Jaakko, and Robert Plonsey. 1995. *Bioelectromagnetism: Principles and Applications of Bioelectric and Biomagnetic Fields*. *Bioelectromagnetism: Principles and Applications of Bioelectric and Biomagnetic Fields*. Oxford University Press.

Mamdani, E. H. 1974. “APPLICATION OF FUZZY ALGORITHMS FOR CONTROL OF SIMPLE DYNAMIC PLANT.” *Proceedings of the Institution of Electrical Engineers* 121 (12): 1585–88.

Mardi, Zahra, Seyedeh Naghmeh Miri Ashtiani, and Mohammad Mikaili. 2011. “EEG-Based Drowsiness Detection for Safe Driving Using Chaotic Features and Statistical Tests.” *Journal of Medical Signals and Sensors* 1 (2): 130–37.

Masisi, Lesedi, Fulufhelo V. Nelwamondo, and Tshilidzi Marwala. 2008. “The Effect of Structural Diversity of an Ensemble of Classifiers on Classification Accuracy,” April.

Mason, Steven G, and Gary E Birch. 2003. “A General Framework for Brain-Computer Interface Design.” *IEEE Transactions on Neural Systems and Rehabilitation Engineering : A Publication of the IEEE Engineering in Medicine and Biology Society* 11 (1): 70–85..

Middendorf, M, G McMillan, G Calhoun, and K S Jones. 2000. “Brain-Computer Interfaces Based on the Steady-State Visual-Evoked Response.” *IEEE Transactions on Rehabilitation Engineering : A Publication of the IEEE Engineering in Medicine and Biology Society* 8 (2): 211–14.

Moran, Rosalyn. 2015. “Introduction.” In , 1–14.

Morrell, Martha J. 2011. “Responsive Cortical Stimulation for the Treatment of Medically Intractable Partial Epilepsy.” *Neurology* 77 (13): 1295–1304.

Müller, Klaus-Robert, Charles W Anderson, and Gary E Birch. 2003. “Linear and Nonlinear Methods for Brain-Computer Interfaces.” *IEEE Transactions on Neural Systems and Rehabilitation Engineering : A Publication of the IEEE Engineering in Medicine and Biology Society* 11 (2): 165–69.

Murray Sherman, S., and R.W. Guillery. 2001. *Exploring the Thalamus*. *Exploring the Thalamus*. Elsevier.

Na, Shi, Liu Xumin, and Guan Yong. 2010. “Research on K-Means Clustering Algorithm: An Improved k-Means Clustering Algorithm.” In *2010 Third International Symposium on Intelligent Information Technology and Security Informatics*, 63–67. IEEE.

Neuper, C, and G Pfurtscheller. 2001. “Event-Related Dynamics of Cortical Rhythms: Frequency-Specific Features and Functional Correlates.” *International Journal of Psychophysiology* 43 (1): 41–58.

Nevado-Holgado, Alejo J, Nicolas Mallet, Peter J Magill, and Rafal Bogacz. 2014. “Effective Connectivity of the Subthalamic Nucleus-Globus Pallidus Network during Parkinsonian

Oscillations." *The Journal of Physiology* 592 (Pt 7): 1429–55.

Nevado-Holgado, Alejo J, Frank Marten, Mark P Richardson, and John R Terry. 2012. "Characterising the Dynamics of EEG Waveforms as the Path through Parameter Space of a Neural Mass Model: Application to Epilepsy Seizure Evolution." *NeuroImage* 59 (3): 2374–92.

Omerhodzic, Ibrahim, Samir Avdakovic, Amir Nuhanovic, and Kemal Dizdarevic. 2013. "Energy Distribution of EEG Signals: EEG Signal Wavelet-Neural Network Classifier," July.

Osorio, Ivan, Mark G. Frei, and Steven B. Wilkinson. 1998. "Real-Time Automated Detection and Quantitative Analysis of Seizures and Short-Term Prediction of Clinical Onset." *Epilepsia* 39 (6): 615–27.

Paenke, I., and J. Branke. 2006. "Efficient Search for Robust Solutions by Means of Evolutionary Algorithms and Fitness Approximation." *IEEE Transactions on Evolutionary Computation* 10 (4): 405–20.

Parvez, Mohammad Zavid, and Manoranjan Paul. 2014. "Epileptic Seizure Detection by Analyzing EEG Signals Using Different Transformation Techniques." *Neurocomputing* 145 (December): 190–200.

Pastor, Maria A, Julio Artieda, Javier Arbizu, Miguel Valencia, and Jose C Masdeu. 2003. "Human Cerebral Activation during Steady-State Visual-Evoked Responses." *The Journal of Neuroscience : The Official Journal of the Society for Neuroscience* 23 (37): 11621–27.

Pathirage, Indika, Karan Khokar, Elijah Klay, Redwan Alqasemi, and Rajiv Dubey. 2013. "A Vision Based P300 Brain Computer Interface for Grasping Using a Wheelchair-Mounted Robotic Arm." In *2013 IEEE/ASME International Conference on Advanced Intelligent Mechatronics*, 188–93. IEEE.

Pelli, D G. 1997. "The VideoToolbox Software for Visual Psychophysics: Transforming Numbers into Movies." *Spatial Vision* 10 (4): 437–42.

Petersen, Sabine, Ulf Zimmermann, Christian Schmidt, Lars Schwabe, Mareike Warkentin, and Stefan J Teipel. 2014. "Linking a Neural Mass Model with a 3D Model of the Human Brain to Reproduce EEG Signals." *Biomedizinische Technik. Biomedical Engineering* 59 (3): 231–40.

Pfurtscheller, G., and F.H. Lopes da Silva. 1999. “Event-Related EEG/MEG Synchronization and Desynchronization: Basic Principles.” *Clinical Neurophysiology* 110 (11): 1842–57.

Pierro, Francesco di, Soon-Thiam Khu, and Dragan A. Savi. 2007. “An Investigation on Preference Order Ranking Scheme for Multiobjective Evolutionary Optimization.” *IEEE Transactions on Evolutionary Computation* 11 (1): 17–45.

Platt, John C., Nello Cristianini, and John Shawe-Taylor. 1999. “Large Margin DAGs for Multiclass Classification.”

Polikar, R. 2006. “Ensemble Based Systems in Decision Making.” *IEEE Circuits and Systems Magazine* 6 (3): 21–45.

Pota, Marco, Massimo Esposito, and Giuseppe De Pietro. 2017. “Designing Rule-Based Fuzzy Systems for Classification in Medicine.” *Knowledge-Based Systems* 124 (C): 105–32.

Purkayastha, Satyajit Sen, V K Jain, and H K Sardana. 2014. “Topical Review: A Review of Various Techniques Used for Measuring Brain Activity in Brain Computer Interfaces.” Vol. 4.

Purves, Dale, George J Augustine, David Fitzpatrick, Lawrence C Katz, Anthony-Samuel LaMantia, James O McNamara, and S Mark Williams. 2001. “Excitatory and Inhibitory Postsynaptic Potentials.”

Qi, Yu, Fei-qiang Ma, Ting-ting Ge, Yue-ming Wang, Jun-ming Zhu, Jian-min Zhang, Xiao-xiang Zheng, and Zhao-hui Wu. 2014. “Abidirectional Brain-Computer Interface for Effective Epilepsy Control.” *Journal of Zhejiang University SCIENCE C* 15 (10): 839–47.

Rao, Rajesh P. N. 2013. “Brain-Computer Interfacing | Artificial Intelligence and Natural Language Processing | Cambridge University Press.” 2013.

Rifkin, Ryan, and Aldebaro Klautau. 2004. “In Defense of One-Vs-All Classification.” *Journal of Machine Learning Research* 5 (Jan): 101–41.

Riquelme, Nery, Christian Von Lucken, and Benjamin Baran. 2015. “Performance Metrics in Multi-Objective Optimization.” In *2015 Latin American Computing Conference (CLEI)*, 1–11. IEEE.

Riza, Lala Septem, Christoph Bergmeir, Francisco Herrera, and José Manuel Benítez. 2015. “Frbs: Fuzzy Rule-Based Systems for Classification and Regression in R.” *Journal of*

Statistical Software 65 (6).

Rokach, Lior. 2010. “Ensemble-Based Classifiers.” *Artificial Intelligence Review* 33 (1–2): 1–39.

Rosso, O.A, M.T Martin, and A Plastino. 2002. “Brain Electrical Activity Analysis Using Wavelet-Based Informational Tools.” *Physica A: Statistical Mechanics and Its Applications* 313 (3–4): 587–608.

Rosso, Osvaldo A., Susana Blanco, Juliana Yordanova, Vasil Kolev, Alejandra Figliola, Martin Schürmann, and Erol Başar. 2001. “Wavelet Entropy: A New Tool for Analysis of Short Duration Brain Electrical Signals.” *Journal of Neuroscience Methods* 105 (1): 65–75.

Rotterdam, A van, F H Lopes da Silva, J van den Ende, M A Viergever, and A J Hermans. 1982. “A Model of the Spatial-Temporal Characteristics of the Alpha Rhythm.” *Bulletin of Mathematical Biology* 44 (2): 283–305.

Sagi, Omer, and Lior Rokach. 2018. “Ensemble Learning: A Survey.” *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery* 8 (4).

Said, Lamjed Ben, Slim Bechikh, and Khaled Ghedira. 2010. “The R-Dominance: A New Dominance Relation for Interactive Evolutionary Multicriteria Decision Making.” *IEEE Transactions on Evolutionary Computation* 14 (5): 801–18.

Salomon, Shaul, Gideon Avigad, Peter J Fleming, and Robin C Purshouse. 2014. “Active Robust Optimization: Enhancing Robustness to Uncertain Environments.” *IEEE Transactions on Cybernetics* 44 (11): 2221–31.

Saraswat, Amit, and Ashish Saini. 2013. “Multi-Objective Optimal Reactive Power Dispatch Considering Voltage Stability in Power Systems Using HFMOEA.” *Engineering Applications of Artificial Intelligence* 26 (1): 390–404.

Savic, Dragan. 2002. “Single-Objective vs. Multiobjective Optimisation for Inte-Grated Decision Support.”

Schapire, Robert E. 1990. “The Strength of Weak Learnability.” *Machine Learning* 5 (2): 197–227.

Schiff, Steven J. 2011. “Neural Control Engineering: The Emerging Intersection between

Control Theory and Neuroscience (Computational Neuroscience): 9780262015370: Medicine & Health Science Books @ Amazon.Com." 2011.

Shannon, C. E. 1948. "A Mathematical Theory of Communication." *Bell System Technical Journal* 27 (3): 379–423.

Sharkey, Amanda J. C. 1999. "Multi-Net Systems." In , 1–30.

Sherman, S. 2006. "Thalamus." *Scholarpedia* 1 (9): 1583.

Shih, Jerry J, Dean J Krusienski, and Jonathan R Wolpaw. 2012. "Brain-Computer Interfaces in Medicine." *Mayo Clinic Proceedings* 87 (3): 268–79.

Späth, Helmuth. 1980. *Cluster Analysis Algorithms for Data Reduction and Classification of Objects*. E. Horwood.

Subasi, Abdulhamit. 2007. "Application of Adaptive Neuro-Fuzzy Inference System for Epileptic Seizure Detection Using Wavelet Feature Extraction." *Computers in Biology and Medicine* 37 (2): 227–44.

Subha, D. Puthankattil, Paul K. Joseph, Rajendra Acharya U, and Choo Min Lim. 2010. "EEG Signal Analysis: A Survey." *Journal of Medical Systems* 34 (2): 195–212.

Subramani, Prabakaran, Rajendra Sahu, and Shekhar Verma. 2006. "Feature Selection Using Haar Wavelet Power Spectrum." *BMC Bioinformatics* 7 (1): 432.

Sur, Shravani, and V K Sinha. 2009. "Event-Related Potential: An Overview." *Industrial Psychiatry Journal* 18 (1): 70–73.

Takagi, Tomohiro, and Michio Sugeno. 1985. "Fuzzy Identification of Systems and Its Applications to Modeling and Control." *IEEE Transactions on Systems, Man and Cybernetics* SMC-15 (1): 116–32.

Takahashi, Fumitake, Fumitake Takahashi, and Shigeo Abe. 2003. "S.: Optimizing Directed Acyclic Graph Support Vector Machines." *IN PROC. OF ANN IN PATTERN RECOGNITION*, 166--170.

Tsujinishi, Daisuke, Yoshiaki Koshiba, and Shigeo Abe. 2004. "Why Pairwise Is Better than One-against-All or All-at-Once." In *IEEE International Conference on Neural Networks - Conference Proceedings*, 1:693–98.

Wang, Xin, Friedrich T Sommer, and Judith A Hirsch. 2011. “Inhibitory Circuits for Visual Processing in Thalamus.” *Current Opinion in Neurobiology* 21 (5): 726–33.

Wang, Yinglin, and Tianrui Li, eds. 2012. *Foundations of Intelligent Systems*. Vol. 122. Advances in Intelligent and Soft Computing. Berlin, Heidelberg: Springer Berlin Heidelberg.

Wendling, F, F Bartolomei, J J Bellanger, and P Chauvel. 2002. “Epileptic Fast Activity Can Be Explained by a Model of Impaired GABAergic Dendritic Inhibition.” *The European Journal of Neuroscience* 15 (9): 1499–1508.

White, J A, M I Banks, R A Pearce, and N J Kopell. 2000. “Networks of Interneurons with Fast and Slow Gamma-Aminobutyric Acid Type A (GABAA) Kinetics Provide Substrate for Mixed Gamma-Theta Rhythm.” *Proceedings of the National Academy of Sciences of the United States of America* 97 (14): 8128–33.

Whittington, M A, R D Traub, N Kopell, B Ermentrout, and E H Buhl. 2000. “Inhibition-Based Rhythms: Experimental and Mathematical Observations on Network Dynamics.” *International Journal of Psychophysiology : Official Journal of the International Organization of Psychophysiology* 38 (3): 315–36.

Woldman, Wessel, and John R. Terry. 2015. “Multilevel Computational Modelling in Epilepsy: Classical Studies and Recent Advances.” In , 161–88. Springer International Publishing.

Wolpaw, Jonathan R., and Elizabeth Winter Wolpaw. 2012. *Brain-Computer Interfaces: Principles and Practice*. *Brain-Computer Interfaces: Principles and Practice*. Oxford University Press.

Wolpaw, Jonathan, and Elizabeth Winter Wolpaw. 2012. *Brain–Computer InterfacesPrinciples and Practice*. Oxford University Press.

Xin-Chun Zhang. 2013. “A Novel Cubature Kalman Filter for Nonlinear State Estimation.” In *52nd IEEE Conference on Decision and Control*, 7797–7802. IEEE.

Xin Yao, and Yong Liu. 1998. “Making Use of Population Information in Evolutionary Artificial Neural Networks.” *IEEE Transactions on Systems, Man and Cybernetics, Part B (Cybernetics)* 28 (3): 417–25.

Xiufen Zou, Yu Chen, Minzhong Liu, and Lishan Kang. 2008. “A New Evolutionary

Algorithm for Solving Many-Objective Optimization Problems." *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)* 38 (5): 1402–12.

Xu, Rui, and Donald Wunsch. 2005. "Survey of Clustering Algorithms." *IEEE Transactions on Neural Networks / a Publication of the IEEE Neural Networks Council* 16 (3): 645–78.

Yi, Jun, Junren Bai, Haibo He, Jun Peng, and Dedong Tang. 2018. "Ar-MOEA: A Novel Preference-Based Dominance Relation for Evolutionary Multi-Objective Optimization." *IEEE Transactions on Evolutionary Computation*, 1–1.

Zadeh, L.A. 1965. "Fuzzy Sets." *Information and Control* 8 (3): 338–53.

Zadeh, Lotfi A. 1973. "Outline of a New Approach to the Analysis of Complex Systems and Decision Processes." *IEEE Transactions on Systems, Man and Cybernetics* SMC-3 (1): 28–44.

Zareian, Elham, Jun Chen, and Basabdatta Sen Bhattacharya. 2016. "A Robust Evolutionary Optimisation Approach for Parameterising a Neural Mass Model." In *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 9887 LNCS:225–34. Springer Verlag.

Zavaglia, Melissa, Laura Astolfi, Fabio Babiloni, and Mauro Ursino. 2006. "A Neural Mass Model for the Simulation of Cortical Activity Estimated from High Resolution EEG during Cognitive or Motor Tasks." *Journal of Neuroscience Methods* 157 (2): 317–29.

Zavaglia, Melissa, Laura Astolfi, Fabio Babiloni, and Mauro Ursino. 2008. "A Model of Rhythm Generation and Functional Connectivity during a Simple Motor Task: Preliminary Validation with Real Scalp EEG Data" 10 (1).

Zavaglia, Melissa, Laura Astolfi, Fabio Babiloni, and Mauro Ursino. 2008. "The Effect of Connectivity on EEG Rhythms, Power Spectral Density and Coherence among Coupled Neural Populations: Analysis with a Neural Mass Model." *IEEE Transactions on Bio-Medical Engineering* 55 (1): 69–77.

Zhou, Ligang, Qingyang Wang, and Hamido Fujita. 2017. "One versus One Multi-Class Classification Fusion Using Optimizing Decision Directed Acyclic Graph for Predicting Listing Status of Companies." *Information Fusion* 36 (July): 80–89.

Zitzler, E., and L. Thiele. 1999. "Multiobjective Evolutionary Algorithms: A Comparative Case Study and the Strength Pareto Approach." *IEEE Transactions on Evolutionary*

Computation 3 (4): 257–71.

Zitzler, Eckart, and Eckart Zitzler. 1999. “Evolutionary Algorithms for Multiobjective Optimization: Methods and Applications.”

Zitzler, Eckart, Eckart Zitzler, Marco Laumanns, and Lothar Thiele. 2001. “SPEA2: Improving the Strength Pareto Evolutionary Algorithm,” 95--100.