Online neuro-fuzzy models for real time flow forecasting

Mohammad Ashrafi

2017

Mohammad Ashrafi. (2017). Online neuro-fuzzy models for real time flow forecasting.
Doctoral thesis, Nanyang Technological University, Singapore.

<http://hdl.handle.net/10356/69917>

<https://doi.org/10.32657/10356/69917>

ONLINE NEURO-FUZZY MODELS FOR REAL TIME FLOW FORECASTING

MOHAMMAD ASHRAFI

SCHOOL OF CIVIL AND ENVIRONMENTAL ENGINEERING

2017

ONLINE NEURO-FUZZY MODELS FOR REAL TIME FLOW FORECASTING

MOHAMMAD ASHRAFI

**SCHOOL OF CIVIL AND ENVIRONMENTAL ENGINEERING
NANYANG TECHNOLOGICAL UNIVERSITY**

**A thesis submitted to the Nanyang Technological University in partial fulfillment
of the requirement for the degree of Doctor of Philosophy**

2017

ACKNOWLEDGMENTS

It was a great chance in my life to pursue my education under the supervision of two great men, Assoc. Prof. Chua Lloyd H.C. and Assoc. Prof. Quek Hiok Chai; in one of the most beautiful and lovely cities in the world, Singapore.

Firstly, I would like to express my sincere gratitude to my main advisor Assoc. Prof. Chua Lloyd H.C. who offered me the research opportunity. Thanks for his continuous support of my PhD study and related research, for his patience, motivation, and immense knowledge. His guidance helped me in all the time of research and writing of this thesis. I could not have imagined having a better advisor and mentor for my PhD study.

Besides my advisor, I would like to thank my co-supervisor Assoc. Prof. Quek Hiok Chai from Computational Intelligence Lab, School of Computer Science and Engineering, Nanyang Technological University (NTU). His insightful comments and encouragements incited me to widen my research in computational intelligence. Thanks to him for supporting me with different computational codes.

I would also like to thank my supervisor, Assoc. Prof. Qin Xiaosheng for helping me in my last two years of PhD research. His patience and support helped me to finalize my PhD research.

Special thanks to my seniors, Dr. Talei. Amin, Dr. Yu Lan, Dr. Zhou Weigui and others who guided me through this difficult journey.

Thanks to Singapore for supporting me with SINGA Scholarship. Special thanks to NTU for giving me the financial support for my PhD study. Also special thanks to School of Civil and Environmental Engineering (CEE) faculty and staff.

Last but not the least; I would like to thank my family: my beloved wife and my parents for supporting me spiritually throughout PhD studies and my life in general.

TABLE OF CONTENTS

ACKNOWLEDGMENTS	i
TABLE OF CONTENTS.....	ii
ABSTRACT	vi
LIST OF PUBLICATIONS	viii
LIST OF TABLES	ix
LIST OF FIGURES	x
NOMENCLATURE	xiv
LIST OF SYMBOLS	xviii
CHAPTER 1: INTRODUCTION	1
1.1 Background	1
1.2 Aims and Scope.....	6
1.3 Thesis Structure	7
CHAPTER 2: NEURO-FUZZY MODELS IN FLOW/FLOOD FORECASTING .	8
2.1 Introduction	8
2.2 Adaptive-Network-based Fuzzy Inference System (ANFIS) and Earlier NFMs	10
2.3 Other Neuro-Fuzzy Models.....	14
2.4 Discussion	16
CHAPTER 3: METHODOLOGY, STUDY AREA AND DATA USED.....	18
3.1 Introduction	18
3.2 Artificial Neural Networks	18

3.3	Fuzzy Inference	22
3.3.1	Mamdani FIS	25
3.3.2	Takagi-Sugeno-Kang (TSK) FIS.....	26
3.4	Neuro-Fuzzy Models	28
3.4.1	Neuro-Fuzzy Model Architecture	30
3.4.2	Clustering	32
3.4.3	Parameter Learning	34
3.5	GSETSK.....	35
3.5.1	GSETSK Structure	36
3.5.2	GSETSK Structure Learning.....	38
3.5.3	GSETSK Parameter Learning	43
3.6	Study Area and Data.....	47
3.6.1	Klippan_2 Catchment.....	47
3.6.2	Lower Mekong River	47
3.7	Evaluation Indices	51
3.7.1	Goodness of Fit Indices	51
3.7.2	Threshold-based Indices	52
3.8	Benchmarks	52
3.8.1	Unified Runoff Basin Simulation Model	52
3.8.2	HBV Model	54
CHAPTER 4: RULE BASE DYNAMICS OF NEURO-FUZZY MODELS FOR FLOW FORECASTS		55

4.1	Introduction	55
4.2	Methodology	56
4.2.1	Study Area and Data.....	56
4.2.2	Model Inputs.....	57
4.2.3	Procedure	58
4.3	Results and Discussion	60
4.3.1	Input Selection.....	60
4.3.2	Comparison of Results	63
4.3.3	Rule Base Dynamics	67
4.4	CONCLUSIONS	72
CHAPTER 5: FLOW FORECASTING IN BASINS WITH LIMITED DATA		73
5.1	Introduction	73
5.2	GSETSK Parameter Values Adopted.....	74
5.3	Data Used and Evaluation of Results	75
5.3.1	Study Area and Data.....	75
5.3.2	Model Comparisons	76
5.4	Results and Discussion	77
5.4.1	Forecasting in Basins without Prior Data (Scenario 1)	77
5.4.2	Forecasting in Basins with Limited Data (Scenario 2).....	81
5.4.3	GSETSK Rule Base and Interpretability	83
5.5	Conclusion.....	90
CHAPTER 6: SPARSE FUZZY RULE-BASE FOR FLOW FORECASTING		91

6.1	Introduction	91
6.2	Fuzzy Interpolation and Extrapolation	92
6.3	Proposed FIER-GSETSK Model.....	97
6.4	Results	104
6.4.1	Rainfall-Runoff	104
6.4.2	River Routing	109
6.5	Conclusion.....	114
CHAPTER 7: CONCLUSION		115
7.1	Summary	115
7.2	Future Research Directions	116
REFERENCES		117

ABSTRACT

Forecasting with limited data or sparse data are two main challenges needed to be addressed. Data should be representative of the system under consideration when forecasting with traditional neuro-fuzzy models (NFM); the condition which is not met in case of forecasting with limited data. Also traditional NFMs cannot handle sparse rule base; the condition which is not suitable for modeling the low-frequency events.

Generic Self-Evolving Takagi-Sugeno-Kang or GSETSK is a state-of-the-art NFM with specific capabilities due to its specific clustering, rule pruning and optimization mechanisms. The fully online property allows the model to be used independent of historical data. Also its dynamic expanding-shrinking structure makes its rule base dynamic, compact, up-to-date and interpretable. GSETSK was run fully-online from the 1st sample onwards for two real time hydrologic forecasting problems in this study. The two commonly encountered problems; rainfall-runoff and flow routing modeling represent different modelling complexities. The data used belongs to two totally different datasets including a catchment in Sweden and Lower Mekong River in South-east Asia. In this research firstly the capability of GSETSK in forecasting with large datasets were critically examined. GSETSK was evaluated against traditional NFMs including the local NFM, DENFIS (Dynamic Evolving Neural-Fuzzy Inference System) and global NFM, ANFIS (Adaptive-Network-Based Fuzzy Inference System). Also the rule base dynamics of GSETSK were evaluated in relation to the complexity of the problem. Secondly the fully-online property of GSETSK was examined in forecasting with limited data e.g. forecasting in basins with limited data. Eventually for handling the sparse data, the Fuzzy Interpolative-Extrapolative Reasoning (FIER) was developed and imbedded in GSETSK.

The overall findings of this research are as follows.

- i) In case of large datasets, GSETSK was able to provide comparable results to traditional NFMs, ANFIS and DENFIS. However GSETSK does not need any knowledge of data upper/lower bound plus that its rule base was more compact, up-to-date and interpretable comparing to DENFIS. Also GSETSK needs less training data and time compared to ANFIS. GSETSK was found to be more suited for the complex problem of rainfall-runoff forecasting comparing to river routing problem.
- ii) In case of limited data, e.g. forecasting in basins with limited data, GSETSK was found to be a good candidate. In addition it provided a compact, up-to-date rule base which was easily interpretable and could be related to the physical process under consideration.
- iii) The sparse data was handled by proposed FIER algorithm. The new model, FIER-GSETSK appeared to model the infrequent events better than GSETSK when evaluated based on threshold-based indices.

In all experiments, the results of GSETSK were comparable to the physically-based models currently in use for each dataset. The two datasets used in this study are representative of rainfall-runoff and river routing problem. Extensive datasets should be used in other researches in future and also other hydrologic applications to examine its generalization ability. Also boosting the model with online input selection and online optimization of the clusters are recommended as the future research direction.

LIST OF PUBLICATIONS

- Ashrafi, M., Chua, L.H.C., Quek, C. and Qin, X. A fully-online Neuro-Fuzzy model for flow forecasting in basins with limited data. 2016. <http://dx.doi.org/10.1016/j.jhydrol.2016.11.057>
- Ashrafi, M., Chua, L.H.C., Quek, C., and Qin, X., Rule base dynamics of neuro-fuzzy models for flow forecasting. J Hydrology. Under review.
- Ashrafi, M., Chua, L.H.C., Quek, C., and Qin, X., Flow forecasting in sparse fuzzy rule bases. Under preparation.

LIST OF TABLES

Table 3.1: Comparison of NFMs	32
Table 4.1: Data used by different NFMs	59
Table 4.2: MCES results for rainfall-runoff application (only positive values are shown)	60
Table 4.3: Cross validation results for rainfall-runoff application	61
Table 4.4: MCES results for river routing application (only positive values are shown)	61
Table 4.5: Results of cross validation procedure for different model input scenarios for river routing forecasts.	62
Table 5.1: GSETSK parameter setting for rainfall-runoff and river routing forecast applications	74
Table 5.2: Data used for the training/testing of data-driven models (excluding GSETSK) in Scenario (2): Basins with limited data	77
Table 5.3: (a) Comparison of GSETSK vs. Naïve model for rainfall-runoff forecast application (1979-1980). (b) Comparison of GSETSK vs. HBV model for rainfall-runoff forecast (1980).....	78
Table 5.4: Performance of GSETSK against traditional models for rainfall-runoff application.	81
Table 5.5: Performance of GSETSK against traditional models for river forecast application	82
Table 6.1: FIER strategy adopted in FIER-GSETSK for rainfall-runoff modeling	108
Table 6.2: FIER strategy taken for different dimensions when using FIER-GSETSK for 1-3-5 day ahead forecasting in Mekong	113

LIST OF FIGURES

Figure 1.1: A flood forecasting system (adopted from Sene (2008))	1
Figure 1.2: Time delays when issuing a warning for a rainfall runoff model (adopted from Sene (2008))	2
Figure 1.3: Different types of models	3
Figure 1.4: Taxonomy of NFMs	4
Figure 3.1: Structure of the biological neuron (adopted from Hagan et al. (1996))	19
Figure 3.2: Model of the artificial neuron (adopted from Demuth et al. (2009))	20
Figure 3.3: (a) linear, (b) radial basis, (c) hyperbolic tangent sigmoid transfer functions (adopted from Hagan et al. (1996)).....	20
Figure 3.4: Three layer feedforward ANN (adopted from Demuth et al. (2009)) ...	21
Figure 3.5: Membership as (a) binary or (b) fuzzy (degree of membership)	23
Figure 3.6: (a) Gaussian and (b) Triangular MFs	24
Figure 3.7: Fuzzy logic operations on three Gaussian MFs for OR and AND operators.....	24
Figure 3.8: An example of a Mamdani type FIS	26
Figure 3.9: An example of a TSK type FIS	28
Figure 3.10: General architecture of NFMs; in a static NFM, model structure is fixed (full line). In a dynamic NFM, the model structure can evolve during training as indicated by the dashed lines.....	31
Figure 3.11: (a) SC method applied for 2-D input space, (b) Density measure for an arbitrary data point effective at radius r_a , where d is the Euclidean distance	33

Figure 3.12: ECM Clustering mechanism for a newly arrived data point in 2-D input space	34
Figure 3.13: GSETSK architecture	38
Figure 3.14: GSETSK knowledge-base management	42
Figure 3.15: GSETSK flowchart.....	46
Figure 3.16: Klippan_2 Catchment (adopted from Talei (2013)).....	48
Figure 3.17: Mekong River watershed (adopted from Mekong River Commission (2011)).....	49
Figure 3.18: Lower Mekong River watershed (adopted from Mekong River Commission (2011))	50
Figure 4.1: Evaluation indices for different models for rainfall-runoff application. (a) Goodness of fit indices, and (b) Threshold-based indices.	65
Figure 4.2: Results of 1 day ahead rainfall-runoff forecasts.....	65
Figure 4.3: Evaluation indices for river routing application. (a) 1 day ahead, (b) 3 days ahead, (c) 5 days ahead.....	66
Figure 4.4: (a.) Time series of discharge and (b.) Change in the number of rules and yearly FAR during the model operation for different NFMs in the rainfall-runoff forecast application.	69
Figure 4.5: (a.) Time series of water level and (b.) Change in the number of rules and yearly FAR during model operation for different NFMs in the river routing forecast application.	71
Figure 5.1: Comparison of GSETSK and HBV model results for rainfall-runoff forecast application for 1980	79
Figure 5.2: Comparison of GSETSK, URBS and Naïve models for river routing forecast application	79

Figure 5.3: Result of river routing forecast application for Lower Mekong River catchment by (a) GSETSK and (b) URBS.....	80
Figure 5.4: Knowledge-base of different input dimensions at the end of forecasting with DENFIS and GSETSK fuzzy neural modeling (rainfall-runoff forecast).....	85
Figure 5.5: Knowledge-base of different input dimensions at the end of forecasting with DENFIS and GSETSK fuzzy neural modeling (river routing forecast).....	85
Figure 5.6: Rule dynamics of GSETSK for 1 day ahead rainfall-runoff forecast application.....	86
Figure 5.7: Rule dynamics of GSETSK for 1 day ahead river routing forecast application.....	89
Figure 6.1: Coarse-grain-represent of FRB in GSETSK	94
Figure 6.2: Fine-grain-represent of FRB in FIER-GSETSK	94
Figure 6.3: An example of FRI (Closed form FRI) (a) Interpolation procedure, (b) Interpolation results (adapted from Yang and Shen (2013))	96
Figure 6.4: Modifications (highlighted in blue) to GSETSK original flowchart.....	98
Figure 6.5: The linear output space in a TSK NFM and its parameters (2-D input space example).....	99
Figure 6.6: The proposed FIER (2-D input space example).....	102
Figure 6.7: Fuzzy rule: (a), (b) interpolation and (c), (d) extrapolation.	103
Figure 6.8: FIER-GSETSK forecasting results for rainfall-runoff modeling	105
Figure 6.9: General goodness of fit indices for rainfall-runoff modeling (Years 1979-1980).....	105

Figure 6.10: General goodness of fit indices for rainfall-runoff modeling (Year 1980)	106
Figure 6.11: Comparing threshold based indices of different models for rainfall-runoff modeling (Years 1979-1980)	106
Figure 6.12: Comparing threshold based indices of different models for rainfall-runoff modeling (Year 1980)	107
Figure 6.13: Comparison of FIER-GSETSK with measured data for (a.) 2010, (b.) 2011.....	110
Figure 6.14: General goodness of fit indices for (a) 1 day ahead, (b) 3day ahead, (c) 5 day ahead forecasting in Mekong River	111
Figure 6.15: 1-3-5 day ahead forecasting results of different models for Mekong River.....	112

NOMENCLATURE

AARE	Average Absolute Relative Error
ACF	Auto Correlation Function
ANFIS	Adaptive-Network-Based Fuzzy Inference System
ANGIS	Adaptive Neuro-GA Integrated System
ANN	Artificial Neural Network
ANN-GA	Genetic Algorithm-based Artificial Neural Network
AR	Autoregressive
ARIMA	Auto-Regressive Integrated Moving Average
ARMA	Autoregressive Moving Average
ARX	Auto-regressive model with exogenous inputs
BPNN	Back-Propagation Neural Network
CCF	Cross Correlation Function
CE	Coefficient of Efficiency
CFNN	Counterpropagation fuzzy-neural network
CI	Computational Intelligence
CRI	Compositional Rule of Inference
CSI	Critical Success Index
CV	Cross Validation
DDM	Data Driven Model
DENFIS	Dynamic Evolving Neural-Fuzzy Inference System

ECM	Evolving Clustering Method
FA	False Acceptance
FAR	False Alarm Ratio
FFM	Flood Forecasting Model
FFNN	Feedforward Neural Network
FFS	Flood Forecasting System
FI	Fuzzy Inference
FIER	Fuzzy Interpolative/Extrapolative Reasoning
FIER-GSETSK	Fuzzy Interpolating and Extrapolating Reasoning-GSETSK
FIS	Fuzzy Inference System
FNN	Fuzzy Neural Network
FR	False Rejection
FRB	Fuzzy Rule Base
FRBS	Fuzzy-Rule-Based System
FRI/E	Fuzzy Rule Interpolation/Extrapolation
GA	Genetic Algorithm
GMF	Gaussian Membership Function
GP	Genetic Programing
GSETSK	Generic Self-Evolving Takagi-Sugeno-Kang
HBV	Hydrologiska Byråns Vattenbalansavdelning
KWM	Kinematic Wave Model
LFC	Low-flow Criteria

LSE	Least Square Estimation
MAE	Mean Absolute Error
MAPE	Mean Absolute Percentage Error
MARE	Mean Absolute Relative Error
MBE	Mean Bias Error
MCES	Monte Carlo Evaluative Selection
MF	Membership Function
MI	Mutual Information
MLR	Multiple Linear Regression
MRC	Mekong River Commission
MSGC	Multidimensional-Scaling Growing Clustering
NFM	Neuro-Fuzzy Model
NFS	Neuro-Fuzzy System
NGO	Neuro-Genetic Model
NNARX	Neural Network Auto Regressive with exogenous input
NS	Noise to Signal Ratio
NTU	Nanyang Technological University
NRMSE	Normalized root mean square error
PACF	Partial Autocorrelation Function
PDE	Partial Differential Equations
PFC	Peak Flow Criteria
PE	Peak Error

POD	Probability of Detection
R	Coefficient of Correlation
R^2	Coefficient of Determination
RMAE	Relative mean absolute error
RMSE	Root Mean Square Error
RPE	Relative Peak Error
SC	Subtractive Clustering
SEE	Standard Error Estimate
SFMI	Sparse Fuzzy Model Identification
SMHI	Swedish Meteorological and Hydrological Institute
SWMM	Storm Water Management Model
SVM	Support Vector Machine
TA	True Acceptance
TS	Threshold Statistics
TSK	Takagi-Sugeno-Kang
URBS	Unified Runoff Basin Simulation

LIST OF SYMBOLS

c_{i,j_i}	Center of GMF
σ_{i,j_i}	Width of GMF
x_i	Input value
i	Input dimension index
j	GMF index
k	Rule number Index
r_k	Spatial firing strength
ω_k	Average forward firing strength
ρ	ρ^{th} rule which has the maximum firing strength
ε	Largest matching degree
η	Degree of overlap between two GMF
$WP(t)$	Willingness Parameter
$P_k(t)$	Fuzzy rule potential
$e_k(t)$	Error of node K at time t

$r_k^{back}(t)$ Backward firing strength of node K at time t

λ Forgetting factor

CHAPTER 1: INTRODUCTION

1.1 Background

Flood is a covering by water of land not normally covered by water (European Union Flood Directive, 2007). Floods place life and properties at risk and there is therefore attention focused against this natural disaster. Nowadays flood forecasting systems (FFSs) (Figure 1.1) are widely used to provide early flood warning in order that emergency response can be provided in a timely manner.

Figure 1.1: A flood forecasting system (adopted from Sene (2008))

Common with most FFSs is that the required inputs are gathered from sensors (measured data) or from models (model data) and are fed to the flood forecasting

models (FFMs) after which forecasting results would be disseminated for emergency response. An ideal FFM should be able to provide accurate forecasts based on the available data in a timely manner, in order that sufficient lead time (Figure 1.2) is available.

Figure 1.2: Time delays when issuing a warning for a rainfall runoff model (adopted from Sene (2008))

Flood forecast models can be categorized based on the consideration of physical process in the system. These include White Box, Grey Box and Black Box models (Figure 1.3). White box models or distributed physically-based models (Refsgaard, 1996) describe clearly the governing physical processes, over the model area partitioned into a set of computational cells. Partial Differential Equations (PDEs) would be used to describe mass, momentum and energy conservation. In this model, the boundary and initial conditions should be set and parameters should be estimated using field or lab experiments. Example of these models are MGB-IPH (Collischonn et al., 2007), LISFLOOD (De Roo et al., 2000) and TOPKAPI (Liu et al., 2005). Grey box or lumped models are a type of model which divides a catchment into hydrologic units. In this model, storages are considered and their

recharging and depletions model different processes in the system. The mathematical requirements are reduced to simpler physical equations, sometimes in the form of a mass balance. The model outputs are an average of the system behavior and ‘automated fitting techniques are often used to provide an initial estimate for the parameters’ (Sene, 2010). HBV (Bergstrom, 1995) and PDM (Moore, 2007) are some examples of this category of models. Black box models do not rely on a physical description of the system. Rather, black box models depend on the concurrent introduction of inputs and outputs. Refsgaard (1996), divides these types of the model in to three groups including empirical hydrological methods (e.g. unit hydrograph method) (Sherman, 1932), statistically based methods (e.g. auto-regressive integrated moving average or ARIMA) (Box et al., 2008) and hydroinformatics based models which include data driven models (DDMs).

Comparing amongst these three categories of FFMs, it is obvious that the relative advantage of each category of model over another depends on considerations such as time, cost, accuracy, and spatial and temporal resolution. In view of this, distributed physically-based models are able to make a detailed view of the system but are time consuming and expensive due to their data requirements. Indeed, sometimes, the data required are not available. On the other hand, DDMs are ‘parsimonious, run times are fast, and models are tolerant to data loss’ (Seo and Demargne, 2008).

Figure 1.3: Different types of models

Solomatine et al., (2008) discussed DDM as the area of hydroinformatics which has focused on computational intelligence (CI) and machine learning (ML) methods. Among these two methods, CI attempts to mimic the learning characteristics of the human brain including features such as robustness, fault tolerance, flexibility, highly parallel and nonlinearity and its ability to handle imprecise, noisy or inconsistent information and are characterized in neuro-fuzzy models (NFM). This class of DDM is a synergism of artificial neural networks (ANN) and fuzzy inference system (FIS) and is also referred to as neuro-fuzzy system (NFS) or fuzzy neural network (FNN). Much research has been done on this area in the last two decades resulting in the evolution of NFM from primitive static (such as the Adaptive-Network-based Fuzzy Inference System or ANFIS (Jang, 1993)) to self-organizing (such as the Dynamic Evolving Neural-Fuzzy Inference System or (DENFIS) (Kasabov and Song, 2002)) to self-evolving versions (such as Generic Self-Evolving Takagi-Sugeno-Kang or GSETSK (Nguyen, 2012; Nguyen et al., 2015)) (Figure 1.4). This research uses the state of the art self-evolving NFM.

Figure 1.4: Taxonomy of NFMs

As in most DDMs, NFM create a mapping between inputs-output spaces based on the concurrent representation of inputs-output pairs to the model in the training phase. This mapping then can be used for modeling the output in the testing phase. NFMs learning schema can mainly be divided into batch or incremental (local) learning. In batch mode, the model is trained using batch or historical data while in incremental mode, training is carried out with the arrival of each new data sample. In batch learning, the new arrival data does not contribute to further learning or refinement of the model while in incremental learning mode, the model is refined

with the arrival of individual data points. This characteristic is particularly useful for online applications, since data is fed incrementally. ANFIS is probably the most common NFM flood forecast which adopts the batch learning schema and suffers from the above-mentioned downside of batch learning. Even for the rare cases of the pseudo-incremental version of ANFIS (Rath et al., 2013), besides increase in runtime, ANFIS structure is fixed or static which makes the model unresponsive to changes. This has given rise to the adoption of an incremental NFMs such as DENFIS in forecasting problems recently (Talei et al., 2013) where a dynamic structure tries to cope better with the nature of floods. In this type of model, the structure of the model evolves by training the model at every arrival of input/output data. Despite its merits however, DENFIS suffers from the following drawbacks: (i) Upper/lower bound of data are required for the normalization process. This may cause serious problems for DENFIS in the case that arrival data is outside the range of trained data which is high especially for extreme events. Also this may cause some difficulties in case of forecasting with limited datasets e.g. forecasting in basins with limited dataset. (ii) The number of fuzzy rules increases monotonically during the lifetime of DENFIS which may generate irrelevant rules. This eventually would deteriorate the accuracy of forecast. Also lots of highly overlapping rules are created which deteriorates its interpretability.

To address the above-mentioned challenges, the Generic Self-Evolving Takagi-Sugeno-Kang (GSETSK) model (Nguyen, 2012; Nguyen et al., 2015) was used for this research. The details of this model are provided in later chapters; however, some of its salient features and how it overcomes the inherent weakness of DENFIS are mentioned briefly here. Firstly, the clustering method used in GSETSK which is known as Multidimensional-Scaling Growing Clustering or MSGC does not require normalization and hence the data lower/upper bound is not required. Secondly, GSETSK uses a rule pruning algorithm which guarantees a more up-to-date rule base by keeping the rule-base relevant, compact and up-to-date. In spite of these advantages, the following two challenges have been identified with the use of GSETSK and are the topics for this study. The first one is the loss of valuable rules associated with low frequency events (floods). This is due to the

operation of rule pruning where the rules which are not active would be pruned. Rule pruning based on historical measures may not be suitable for forecasting extreme events which occur infrequently. Secondly GSETSK relies on wide Gaussian membership functions (GMFs) which makes for a coarse-representation and a uniformly distributed fuzzy rule-base (FRB). This means there are a limited numbers of rules which can be used for modeling. If these rules are mainly tuned by the high-frequency events, the model may not be adequate for the low-frequency events.

1.2 Aims and Scope

The aims of this thesis can be summarized as follows:

- (i) To evaluate the advancements in NFMs for flow/flood forecasting. This includes the assessment of local versus global learning strategies and the study of the importance of the rule base dynamics in relation to hydrologic forecasts.
- (ii) To assess the applicability of GSETSK as an online forecast model in basins without or with limited prior data
- (iii) To develop an interpolation/extrapolation solution which can handle sparse FRB. This involves the development of a fine-representation FRB which is better able to model infrequently occurring extreme events such as floods

Two datasets from different geographical locations were used in this research. The first dataset is from a Swedish Catchment and is for a rainfall-runoff model study while the second dataset is from Mekong River and was used for river routing forecasts. The use of these two datasets tests the model for the two types of hydrological forecast problems commonly encountered. One-day ahead forecast of runoff was studied with the Swedish catchment data while multi-step-ahead (up to 5-days) water level forecasts was studied with the Mekong River data. Different evaluation indices including the general goodness-of-fit and threshold-based indices

were used in this study. Lastly, GSETSK model results were benchmarked against results obtained from physically based models.

1.3 Thesis Structure

This thesis includes 7 chapters. Chapter 1 introduces the thesis and includes background information about flood forecasting and different type of models used for forecasts. The use of NFMs is elaborated. Chapter 2 provides an overall review of past research that has been conducted using NFMs for flow/flood forecasting. Issues and research gaps are mentioned. Further review is provided in other chapters in the context of the specific topics studied. Chapter 3 outlines the basic elements of NFMs with details of GSETSK provided. The data used, evaluation indices and benchmarks are also presented there. In Chapter 4, the advancements in the NFMs from the static to self-organizing and self-evolving NFMs are investigated. ANFIS, DENFIS and GSETSK are the models adopted as representative of static, self-organizing and self-evolving NFMs, respectively, and the role of rule-base dynamics is investigated in relation to problem (rainfall-runoff and river routing) complexity. In Chapter 5, application of GSETSK in studies with limited datasets is provided. This chapter highlights the fully online application of NFMs where little or no prior data is available. Chapter 6 proposes methods handling sparse FRB. Results from an adaption of GSETK or GSETSK with fuzzy interpolative/extrapolative reasoning (FIER-GSETSK) are presented. In Chapter 7, the conclusions of this research are provided and directions for future works are presented.

CHAPTER 2: NEURO-FUZZY MODELS IN FLOW/FLOOD FORECASTING

2.1 Introduction

Different types of flood forecasting models (based on the physical process in the system under consideration) and their pros and cons were explained in Section 1.1. As discussed earlier, among the models, CI techniques has their specific advantages as robustness, fault tolerance, flexibility, highly parallel and nonlinearity and its ability to handle imprecise, noisy or inconsistent information. There are lots of studies using the CI techniques such as ANN and FIS in flood forecasting. ASCE Task Committee on the application of ANNs in hydrology had a comprehensive study as presented in Govindaraju (2000) and Govindaraju and Artific (2000). During the last decade, different types of ANNs and their various aspects have been discussed when being used in different flood forecasting studies. Kang et al. (2006) used ANN solely for flood forecasting in the Republic of Korea. Lu et al. (2006) applied an integration of back-propagation neural network (BPNN) and transfer function for the thyphoons in Taipei, Taiwan. Chang et al. (2007) developed the enforced self-organizing map (ESOM) coupled with a linear regression output layer for flood forecasting of Da-Chia River, Taiwan. Han et al. (2007) investigated using bootsrtap technique in the improvement of ANN for flood forecasting in Bird Creek in the USA.. Leahy et al. (2008) developed a new ANN optimization technique combining the tabu search and simulated annealings for flood forecasting in Ireland. Chang et al. (2009) proposed a radial basis function network (RBFN) which automatically allocates the network structure and does the the search for optimal parameters. They successfully used the model for event-based flood forecasting in Lan-Young River, Taiwan. Chen et al. (2010) developed Decision Group Back-Propagation Network (DGBPN) which selects the best model by comparing the spatial and temporal silmilarities of data patterns. They used this model for flood forecasting in Wu-Shi watershed in Taiwan. Chen et al. (2013b) proposed a recurrent neural network to cope with the lack of current true values at every time step in real time forecasting. Also there are some studies discovering the different

aspects of FIS for flood forecasting. Liong et al. (2000) worked on the interpretability of the model by using fuzzy membership functions on the input/output spaces after clustering data using Kohonen neural network. Nayak et al. (2005a) used a subtractive clustering method for choosing the fuzzy membership functions and Least Square Estimation (LSE) for optimizing the output spaces. Nayak (2010) used subtractive clustering method for determining the membership functions combined by LSE for parameter optimization. Zhang et al. (2012) used the association rule analysis to reduce the number of fuzzy membership function to more necessary ones. They used TSK FIS with membership functions determined manually. Chen et al. (2013a) proposed an evolutionary fuzzy inference model including fuzzy rules, Genetic Programming (GP) for consequent part and genetic algorithm (GA) for optimization of fuzzy membership positions.

Although the ANNs and FISs showed the promising results, there are several disadvantages when using these models. For example ANNs are black box models which do not provide any human interpretable information. Also the prior-knowledge can not be imbedded in ANNs and its optimal parameters is not known. Also in case of FIS, the uncertainty of the clusters position and spread themselves increase the uncertainties. Here is the point that synergism of ANN and FIS makes NFM which brings the advantages of two models while removing their disadvantages. Many different studies have been conducted using NFMs for flow/flood modeling/forecasting. These studies have been conducted at a variety of scales and in different geographical locations. These range from a 25 m² experimental set up in Nanyang Technological University (NTU), Singapore (Talei et al., 2010b) to a 144,500 km² catchment in China (Wang et al., 2009). Also the modeling time-step ranges from minutes to a month and forecast lead times range from 1-step to multi-step ahead forecasts. Furthermore, some of these studies are event-based while others are continuous. NFMs have been used for rainfall-runoff or flow routing applications. Almost all of these studies use a pre-processing step for choosing the best inputs. This includes using expert knowledge, statistical techniques such as Cross Correlation Function (CCF), Auto Correlation Function (ACF), Partial Autocorrelation Function (PACF), Mutual Information (MI), Gamma

test, k-fold Cross Validation (CV), and the analysis of different antecedent inputs and residuals. In these studies general goodness of fit or specific indices have been used for evaluation of the NFM. Also the results of these studies have been evaluated against different benchmarks including physically-based to statistical and data driven models. Most of the earlier papers on NFMs used Adaptive-Network-based Fuzzy Inference System (ANFIS). However, more recently studies have used other types of NFMs. The following two sections attempts to provide a broad review of relevant literature on the application of neuro fuzzy models (and trace their evolution) in flow forecasting, and elucidate general trends/features in the key literature and identify gaps. More in-depth reviews of the literature will be provided in later chapters in the context of specific topics being addressed.

2.2 Adaptive-Network-based Fuzzy Inference System (ANFIS) and Earlier NFMs

The first application of NFM in stream flow prediction seems to be reported in Chang and Chen (2001). In this research a counterpropagation fuzzy-neural network (CFNN) was utilized. This model was used to forecast the flow of Da-cha River in central Taiwan with the catchment area of 514 km². The authors used the lagged precipitation and discharge data for hourly discharge forecasting. The normalized root mean square error (NRMSE), RMSE, mean absolute error (MAE) and relative MAE (RMAE) were used for model evaluation. The result of this study shows the superiority and reliability of this model against ARMAX.

Nayak et al (2004) used ANFIS for daily river flow modeling of the Baitarani River in Orissa State for a watershed 14,218 km² in size. They used the endogenous variables, antecedent river flows as the inputs in this study. Different input combinations were tried in ANFIS model and the residuals were examined to select proper inputs. Also 4-fold cross validation (CV) was used to find the best model. In this study the general goodness-of-fit indices were used for evaluation including RMSE, coefficient of efficiency (CE), and coefficient of correlation (R). These results show the superiority of ANFIS over autoregressive moving average

(ARMA) in terms of general goodness-of-fit indices. Also ANFIS was found to be more accurate than the BPNN model specifically for prediction error distribution indices such as average absolute relative error (AARE) and threshold statistics (TS). In another study, Nayak et al. (2005b) used ANFIS for short-term flood forecasting using rainfall-runoff data for the Kolar Basin, India which has a catchment area of 1,350 km². In this study, different methods including ACF, CCF, and PACF were used for input selection and ANFIS results were compared with ANN and FIS models. The forecasting was done for 1 to up to 6 hours ahead. Finally three type of indices were used including goodness of fit indices (RMSE, CE, R, standard error estimate (SEE), noise to signal ratio (NS), and mean bias error (MBE)), prediction error distribution indices (AARE and TS) and Peak-flow indices (percentage error in predicted peak flow, percent error in total runoff volume, peak flow criteria (PFC) and low-flow criteria (LFC)). The study showed that ANFIS gave better results compared to ANN and FIS for lead times greater than 1 hour.

Chau et al. (2005) used ANFIS and genetic algorithm-based artificial neural network (ANN-GA) for flood forecasting. The authors modeled 1-day-ahead water flow at a station in a channel reach of the Yangtze River. Daily average water flow at the upper station used for this purpose and RMSE and R indices were used for evaluation of the models. Both ANFIS and ANN-GA outperformed the benchmark, a linear regression model. The main drawback however is that ANN-GA has more parameters and needs more computation time. Chen et al. (2006) applied ANFIS in event-based hourly flood forecasting for the Choshui River in central Taiwan (catchment area = 3,157 km²). The authors used the data from 2 river gauge stations and 10 rainfall stations. The travel time between rain gauges and water-level gauges was used for finding the travel time of the flow. Upstream water level and rainfall at different stations were compared with the water level at the downstream station to find the best inputs and the average rainfall of all stations, lagged 5 hours was selected. The results of this study showed that ANFIS gave better performance when compared to BPNN for 1 and 2-step-ahead flood forecasting.

Aqil et al. (2007) used ANFIS for hourly and daily runoff modeling for the Cilalawi River basin (area = 60.17 km²) in West Java Province, Indonesia. The authors used

CCF, ACF, and PACF for selection of best rainfall and runoff data for modeling. The modeling also was done with Levenberg–Marquardt feed forward neural network, and Bayesian regularization feed forward neural network. The evaluation indices used in this study were CE, RMSE, mean absolute percentage error (MAPE). ANFIS model results were better than the two other models.

Firat and Güngör, (2007, 2008) investigated the potential of ANFIS for daily river flow forecasting. The forecasting was done for River Great Menderes in West Turkey, with the area of 24,976 km². Cross correlation analysis was used for selection of the most important lagged inputs. Different combination of inputs were used and results of this study showed that ANFIS performed better than the benchmarks (ANN, multiple linear regression model (MLR) and autoregressive (AR) model) adopted when compared in terms of CE, R, RMSE, and AARE. Remesan et al. (2008) used ANFIS for hourly rainfall-runoff modeling in the 135 km² Brue Basin in the UK. They used the Gamma Test (Evans and Jones, 2002) to select the best input combinations. The results were evaluated using CE, R, RMSE, and MAPE error indices which showed that ANFIS performed better than the Neural Network Auto Regressive with exogenous input (NNARX) model. Mukerji et al. (2009) investigated the performance of ANFIS, ANN, and adaptive neuro-GA integrated system (ANGIS) for 3, 6 and 9 hour ahead flood forecasting. Event-based forecasts was carried out for the 2,882 km² Ajay River Basin in Jharkhand, India using the rainfall-runoff data. No input selection method was adopted, instead all inputs up to 4 days prior were used in the model. RMSE, coefficient of determination (R^2), percentage deviation in peak flow (Dev), and time difference to peak flow (Dep) were used for evaluation. The results showed that ANFIS performed better than ANN in most of cases, although ANGIS generally performed better than ANFIS and ANN.

Wang et al. (2009) applied ANFIS as well as ARMA, ANN, GP and Support Vector Machine (SVM) for forecasting monthly discharge. The forecasting was done for two locations in China. One is Manwan Hydropower in Lancangjiang River with the catchment area of 114,500 km². The other one is Hongjiadu Hydropower on Wujiang River with 9900 km² catchment area. They used ACF and PACF

techniques for choosing the best lagged discharges and used the general indices (CE, R^2 , RMSE, MAPE) for model evaluation. This study showed that ANFIS, GP and SVM achieved the best results in term of all evaluation indices.

Firat and Turan (2010) applied ANFIS for monthly river flow forecasting in Göksu River in Southern Turkey. The authors developed models containing monthly river flow from 1 up to 7 months and tested these models in ANFIS. The results of this study showed that ANFIS produced better results compared to the feedforward neural network (FFNN) and AR models. The goodness-of-fit indices used in the study include CE, RMSE, and R. Rath et al., (2013) applied ANFIS for flood forecasting in the 4,980 km² Narmada basin in India. The authors used CCF, ACF and PACF for choosing the best lagged inputs from the rainfall runoff data. The modeling was done hourly from 1 to up to 6-hour-ahead using ANFIS, ANN and FIS. The general goodness of fit measures, RMSE, CE, MAE, mean absolute relative error (MARE), and specific indices like as TS and peak flow error were used for evaluation. The results of this study showed that ANFIS achieved better results over the other two models.

Talei et al. (2010b) used ANFIS for event-based runoff modeling for a small catchment. They used data which were logged at 15-seconds intervals for a 25 m² experimental station. This study evaluated the rainfall and runoff data of 10 events for discharge modeling of 1 to 8 steps ahead (15 to 120 seconds). The authors used CCF and ACF and proposed different ANFIS models containing rainfall and/or discharge as inputs. Different indices including CE, R^2 , RMSE, MAE (Mean Absolute Error)), peak estimation error and time shift were used to evaluate the best model. The results of this study showed that ANFIS performed better than autoregressive model with exogenous inputs (ARX) for longer lead times. However ARX performed slightly better than ANFIS in peak estimation. In another study, Talei et al. (2010a) applied ANFIS for event-based runoff modeling in a 5.6 km² catchment. The data used in this study were rainfall logged at 5-min intervals. The authors used two steps for finding the best-fit model: (i) they proposed different combinations of lagged rainfall data and checked the error residuals after modeling with ANFIS. (ii) In addition, they used a cross validation method. The models were

evaluated using CE and Relative Peak Error (RPE) as error statistics. The study showed that ANFIS results were comparable to physically-based model Storm Water Management Model (SWMM). In addition ANFIS predicted the peak flow better than SWMM. The extension of this work was reported in (Talei and Chua, 2012) in which the influence of lag time was investigated. They used a CCF and MI in that study to find the best-fit-model. Further improvements were obtained for the ANFIS model developed with the new input selection method. Similarly, it was found that ANFIS results were comparable to SWMM although ANFIS gave better predictions of peak flows compared to SWMM.

Nguyen and Chua (2012) utilized ANFIS for flood forecasting in the Mekong River. They used this model for forecasting the floods at Pakse hydrometric Station in Lao People's Democratic Republic. The authors used CCF for determining the most useful inputs from stations upstream. ACF and PACF were used for choosing the best lag time for the input variables. General goodness-of-fit indices, CE, RMSE, MAE, MPAA on high flow data as well as the whole dataset was used for model evaluation. The results showed that the model achieved the benchmark for 1 to 3-day forecasts but not for 5-day forecasts. In another study, Nguyen et al. (2014) used ANFIS for 1, 3 and 5-day-ahead flood forecasting at three hydrometric stations namely Thakhek, Pakse and Kratie for the Mekong River. The authors used CCF, ACF and PACF for selecting inputs. The ANFIS model achieved better results against other models tested and satisfied benchmark requirements.

2.3 Other Neuro-Fuzzy Models

More recently, there have been a limited number of studies using NFMs other than ANFIS for flow/flood forecasting. These studies have compared their results against the ANFIS model, which up to that time had been the state of the art. Nayak et al. (2007) presented an NFM where its rules are extracted by a hyperellipsoidal fuzzy clustering method and the premise parameters optimized by Levenberg-Marquardt method. This model was used for rainfall-runoff modeling in two basins including Narmada Basin in India and Kentucky Basin in the U.S. The modeling was done

hourly for Narmada and daily for Kentucky. The authors used ACF, PACF, and CCF for selection of the best inputs. Different indices involving CE, RMSE, R, SEE, NS, the relative error, and volume error were used for evaluation. The results of this study showed that the proposed NFM was better than ANFIS when compared using all the evaluation indices. Furthermore, with the new clustering method, fewer rules were generated, resulting in a more transparent and linguistically interpretable model with lesser parameters and computation time.

Chidthong et al. (2009) conducted flood forecast studies with a NFM which used GA for selection of fuzzy rules. The study was conducted with two datasets. The first was from Chiang Mai city in Thailand for which 3, 6, 9, and 12-hrs ahead forecasts was done. The other dataset is from Koriyama City in Japan for which hourly forecasting from 1 to 8-hrs ahead was carried out. The input data used for both datasets were water levels for different stations and rainfall. Inputs were selected based on CCF analysis and general goodness of fit, RMSE, CE and the peak error were used for evaluation. The results of this study showed that the hybrid model and also benchmarks, NGO (neuro-genetic model) and ANFIS predict the water level with satisfactory results, although the hybrid model provided the best peak estimation among all the models.

Talei et al. (2013) used an online NFM known as DENFIS for runoff forecasting in three different basins an artificial catchment with 25 m² area, an urbanized catchment 5.6 km² in area, and the 242.9 km² Klippan_2 catchment in Sweden. Different input selection methods including CCF, MI and CV were used in this study to find the best-fit model. The general goodness-of-fit indices, CE, R², RMSE, MAE, as well as specific peak flow indices Peak Error (PE) were adopted as evaluation measures. The study showed that DENFIS results were comparable or better than the physically-based models used in different catchments. These included Kinematic Wave Model (KWM) for the artificial catchment, SWMM for the urban catchment and Hydrologiska Byråns Vattenbalansavdelning (HBV) for the Swedish catchment. These results were obtained when DENFIS was used in a continuous learning mode, while ANFIS had to be retrained periodically in order to achieve the same performance.

2.4 Discussion

Based on this review of the literature, two key features can be highlighted:

- (i) Almost all studies utilize static NFMs, such as ANFIS, which relies on a batch of data. In these NFMs the structure of the model is fixed and the model cannot be updated incrementally. Incremental updating is a key model feature especially in the modelling of infrequent events such as floods where it is essential to capture local characteristics in the dataset. Global models such as ANFIS rely on batch data for optimization. Although such an approach is suited to model the average behavior of a system, an incremental approach is preferred for modelling infrequent occurrences.
- (ii) Limited studies with dynamic models have been attempted. (Talei et al., 2013) used an online version of DENFIS which is able to update its structure based on the most recent data. However the issue with this model is that it requires prior-knowledge of the upper and lower bounds of the dataset for data normalization. Two drawbacks associated with this requirement are that the model requires substantial historical data for initialization and secondly, the model needs to be retrained when it encounters values that exceed the maximum of minimum values used in the initial training of the model.

In addition, it is also observed that although input selection is an important phase in model preprocessing and two kinds of inputs should be excluded during input selection, i.e. irrelevant inputs and correlated inputs, yet none of the above-mentioned studies exclude correlated inputs. The second observation relates to the indices used for evaluation of model performance. Most studies adopt general goodness of fit indices, although for flow forecasts, threshold based statistics which are important in decision making also should be used. Although many studies using NFMs for flow forecasting have been conducted, however, limitations of current NFMs remain. For real time forecasting, the model should be trained dynamically. This allows the model to be continuously updated or adapted, incrementally, in a fully online manner. Secondly, the model should not rely on data normalization. This negates the need for having a large historical dataset for model initialization.

An advantage of not having this feature is that the model can still be used when datasets are small as is often the case in many practical situations. Lastly, in dealing with floods which is episodic, the input space is heterogeneous, and mechanisms dealing with sparse datasets are needed.

In order to address these issues, the model chosen for this research is GSETSK (Section 3.5), which has a self-evolving structure such that it can adapt its structure based on the most recent data. The model does not require prior-knowledge of data and uses a local optimization technique which updates the model parameters based on the most recent data. An input selection method namely Monte Carlo Evaluative Selection (MCES) (Section 4.2.2) is used in a part of this study as the input selection method. This method results in finding the most relevant input variables which are not correlated to each other. Finally threshold-based indices including critical success index (CSI), probability of detection (POD), false alarm ratio (FAR) (Section 3.7.2) are used in this research to make it more pragmatic. These indices help to have a better understanding of the model in a FFS. Further details of the model are provided in Chapter 3.

CHAPTER 3: METHODOLOGY, STUDY AREA AND DATA USED

3.1 Introduction

Computational Intelligence (CI) was introduced by Zadeh (1994). It attempts to mimic the abilities of the human mind and includes features such as robustness, fault tolerance, flexibility, highly parallel, nonlinearity and capability in handling fuzzy, probabilistic, noisy, or inconsistent information. Computational Intelligence mainly consists of the artificial neural network (ANN), fuzzy logic (FL) and Genetic Algorithm (GA). The combination of these methods makes a powerful tool which can be utilized for real-life problems. In this research a hybridization of ANN and FL or Neuro-Fuzzy Model (NFM) is investigated. A specific NFM, Generic Self-Evolving Takagi-Sugeno-Kang (GSETSK) is mainly utilized in this research.

3.2 Artificial Neural Networks

Artificial neural network (ANNs) are global function approximators which can approximate any function accurately (Hornik, 1991). These non-linear models can learn from data without explicit need of underlying model functional or distributional form (Haykin, 1999). Also the highly-distributed knowledge representation makes it fault-tolerant.

The ANN is inspired by the biological neural system. The basic unit of the neural system is the neuron (Figure 3.1) which consists of three main parts:

- (i) Dendrites which are responsible for receiving the inputs from the environment and transferring them to the cell body after multiplication by their specific weights.
- (ii) Cell Body which is the main part of the neuron where three main operations take place. Firstly the receiving weighted signals from dendrites are summed up

together. Then the bias is added and finally this value would be altered by an activation function to control its amplitude.

- (iii) Axons which are responsible to transfer the output of the cell body to other neurons.

Figure 3.1: Structure of the biological neuron (adopted from Hagan et al. (1996))

Similarly in an artificial neuron (Figure 3.2), the inputs ($P_R, R = 1, \dots, R$), are connected to the neuron body by their corresponding weights ($w_{1,R}, R = 1, \dots, R$).

In the body of the neuron, the weighted inputs and the bias (b) are summed up together. This combination (n) then would be transferred by the activation function (f) to produce the neuron output (a).

Equation 3.1 shows the mathematical formulation of the artificial neuron where $\mathbf{w}$ and $\mathbf{p}$ are the weight and input vectors respectively.

$$a = f(\mathbf{W}\mathbf{p} + b) \quad (3.1)$$

Figure 3.2: Model of the artificial neuron (adopted from Demuth et al. (2009))

There are lots of activations functions which may be used based on the type of problem to be solved. Some of the popular functions are (Figure 3.3):

Hyperbolic tangent sigmoid transfer function

$$a = \frac{1}{1 + e^{-n}} \quad (3.2)$$

Radial basis transfer function

$$a = e^{-n^2} \quad (3.3)$$

Linear transfer function

$$a = n \quad (3.4)$$

where in Eq. (3.2) to (3.4) $n = \mathbf{W} \mathbf{p} + b$ as in Eq. (3.1). Also “a” is the output of the respective activation function

Figure 3.3: (a) linear, (b) radial basis, (c) hyperbolic tangent sigmoid transfer functions (adopted from Hagan et al. (1996))

A neuron has a simple structure and function, but combination of neurons creates a network. When the activation function is linear, the neuron functions as a multiple linear regression (MLR) model. The architecture of a neural network may have different forms. It may have more than one layer which is named as “hidden layer” and different number of the neurons in each layer. ‘The existence of the hidden layer neurons introduces the difference between the linear (e.g., MLR) and the nonlinear model’ (Price and Vojinovic, 2011).

Figure 3.4 shows the architecture of a three-layer ANN. In this figure, the superscripts of each symbol indicate the layer number while the subscripts indicate the neuron number. The weights have two indexes in the subscripts where the first one implies the receiving neuron number and the second indicates the input number for the first layer (iw) or output from the previous layer for hidden nodes (lw).

Figure 3.4: Three layer feedforward ANN (adopted from Demuth et al. (2009))

The mathematical formulation of this network is presented in Eq. (3.5) written in matrix form as:

$$\mathbf{a}^3 = \mathbf{f}^3 (\mathbf{L} \mathbf{W}^{3,2} \mathbf{f}^2 (\mathbf{L} \mathbf{W}^{2,1} \mathbf{f}^1 (\mathbf{I} \mathbf{W}^{1,1} \mathbf{p} + \mathbf{b}^1) + \mathbf{b}^2) + \mathbf{b}^3) \quad (3.5)$$

Where $\mathbf{LW}^{i,j}$ is the weight vector between layer i connected back to layer j . please note that for the 1st layer $i = j = 1$. $\mathbf{p}$ is the input vector, also $\mathbf{b}^i$ and $\mathbf{f}^i$ are the bias vector and activation function for layer i respectively.

The weights and biases in Eq. (3.1) and (3.5) are initially unknown. These would be estimated during supervised/unsupervised learning methods in the training phase. Despite of the advantages mentioned for ANNs there are several issues with them which limit its usage. Firstly ANNs are black box models and there is no way to extract the imbedded knowledge in their structure to a human-interpretable form. Secondly no prior-knowledge can be utilized in the ANNs. Thirdly ANN can not address the stability-plasticity dilemma in which once the ANN training completed, it can not learn the new data. Lastly the optimized structure for an ANN is not known.

3.3 Fuzzy Inference

'Fuzzy inference' (FI) 'is the process of formulating the mapping from a given input to an output using fuzzy logic' (Fuzzy Logic Toolbox™ User's Guide). High-level IF-THEN fuzzy rules are used in FI to model the problem domain. As these rules are intuitive to the human understanding, so the priori-knowledge can be imbedded in FI. In addition, human interpretable knowledge can be extracted from FI. FI mechanism is highly relies on the "fuzzy sets" and "fuzzy logic".

The concept of fuzzy sets was firstly introduced by Zadeh (1965). In a fuzzy set, there is a degree of membership (μ) instead of binary membership (belonging or not belonging to the sets). This concept is illustrated in Figure 3.5 by showing the membership of some data in a 2D space. In classical membership theory, all of the objects belong to the area they are located in, and their membership value is equal to one. In fuzzy membership theory however, there is a degree of membership in the range [0, 1]. Fuzzy sets are created during the process of fuzzification and by using a membership function (MF). Different MFs such as Gaussian and Triangular MFs

(Figure 3.6) are applicable for different types of problems. Equation (3.6) shows the mathematical formulation for a Gaussian MF:

$$f(x; \sigma, c) = \exp\left(\frac{-(x - c)^2}{2\sigma^2}\right) \quad (3.6)$$

where x is the crisp value to be fuzzified and c and σ are the center and the width of the Gaussian MF respectively. Equation (3.7) shows the mathematical formulation for a triangular MFs:

$$f(x, a, b, c) = \begin{cases} 0, & x \leq a \\ (x - a) / (b - a), & a \leq x \leq b \\ (c - x) / (c - b), & b \leq x \leq c \\ 0, & c \leq x \end{cases} \quad (3.7)$$

where a and c indicate the foot and b is the peak of the triangular MF.

Figure 3.5: Membership as (a) binary or (b) fuzzy (degree of membership)

Figure 3.6: (a) Gaussian and (b) Triangular MFs

FI can combine fuzzy sets following a fuzzy logic operation. The most well-known fuzzy logic operations are fuzzy union (OR) and fuzzy intersection (AND). Figure 3.7 illustrates an example of the function of these logic operators on three Gaussian MFs creates. A fuzzy rule is an IF-THEN statement which relates the antecedent to specific consequent in the context of fuzziness. The combination of these rules makes a “fuzzy-rule-based system” (FRBS) which could map the input to output space. Jang (1993) classifies the FIS or FRBS to three types. Types 1, 2 and 3 are known as Tsukamoto (Tsukamoto, 1979), Mamdani (Mamdani and Assilian, 1975) and Takagi-Sugeno-Kang (TSK) (Takagi and Sugeno, 1985), respectively. In the following sections Mamdani and TSK FISs are detailed.

Figure 3.7: Fuzzy logic operations on three Gaussian MFs for OR and AND operators

3.3.1 Mamdani FIS

Mamdani FIS was firstly proposed by Mamdani and Assilian (1975). The FIS can be structured based on the following rules created by the expert knowledge:

Rule 1: IF the precipitation is high AND the temperature is moderate THEN the discharge is high

Rule 2: IF the precipitation is moderate THEN the discharge is moderate

Rule 3: IF the precipitation is low OR temperature is low THEN discharge is low

Now introducing an input pair to the system which contains the precipitation and temperature values the discharge can be estimated. Following is the Mamdani FIS process (Figure 3.8):

- 1- Fuzzification: Crisp input variables are transformed to the fuzzy values by the relative MF.
- 2- Fuzzy logic operation: Fuzzy operators evaluate the antecedent part for each rule and the rule firing strength is created.
- 3- Fuzzy implication: The firing strength is implemented on the MF on the consequent part for each rule.
- 4- Aggregation: The outputs of different rules are summed up.
- 5- Defuzzification: Since the output of the aggregation stage is fuzzy, a specific method like as weighted average is used to transform it to a crisp output.

Figure 3.8: An example of a Mamdani type FIS

3.3.2 Takagi-Sugeno-Kang (TSK) FIS

Takagi-Sugeno-Kang (TSK) or Sugeno was firstly introduced by Sugeno (1985). The first two phases of the Mamdani type are similarly implemented in TSK but the third phase is different (Figure 3.9). In TSK, the consequent is linear or constant which creates a first-order or zero-order TSK, respectively. The general form of the rules is as follows:

Rule i: IF precipitation (P) is ... AND temperature (T) is ... then discharge

$$Z_i = b_{1i}P + b_{2i}T + b_{0i}$$

The linear equation for rule i describes a plane in the high-D space (please refer to Section 6.3 and Figure 6.5 for more details). In this rule z_i is the output of linear function, b_{ji} is the constant parameter (dip of the plane) for input dimension j , and b_{0i} is the distance from the origin of the plane.

In phase three the output of phase two, weight (w_i) is multiplied by the output of the linear or constant function (z_i) for each rule. Aggregation occurs in phase four and finally a weighted averaging is used to extract the output Eq. (3.8):

$$Q = \frac{\sum_{i=1}^i w_i z_i}{\sum_{i=1}^i w_i}, \quad (3.8)$$

Mamdani and TSK can be categorized as “Linguistic” and “Precise” fuzzy models, respectively. The selection of each linguistic or precise fuzzy model as the solver depends on the goal of modeling. An interpretability-accuracy trade off could be used to choose between the fuzzy models as these are incompatible when the system becomes complex (Zadeh, 1973).

The importance of fuzzy modeling has been emphasized by different hydrologists. Mujumdar and Ghosh (2008) state that fuzzy sets are able ‘to address uncertainty due to imprecision, fuzziness or vagueness’. Bardosy (2008) compared linear and nonlinear regression (multiple), nearest neighbor, ANN, and FIS and stated that fuzzy rules ‘can easily cope with missing data and data of poor quality’. These advantages are achievable by using appropriate MF shape and linear function (in TSK FIS) which is led by their parameters which are located in the antecedent and consequent respectively. This point forms motivation for the hybridization of ANN and FI which results in an NFM which is the subject of next section.

Figure 3.9: An example of a TSK type FIS

3.4 Neuro-Fuzzy Models

'Uncertainty in the membership functions themselves may increase model output uncertainty' (Mujumdar and Ghosh, 2008). This is the adverse side of FIS which beside the improper selection of the consequent parameters in TSK FIS, may lead to weak modeling results. To solve this problem, the connectivity and learning ability of ANN are used. An NFM that has seen wide application in the water resource literature is the Adaptive-Network-based Fuzzy Inference System or ANFIS (Jang, 1993). ANFIS which uses TSK in the consequent part is the most famous one and hence ANFIS term refers to this type.

ANFIS is a static or batch model, however, in most forecasting applications, full online capability is needed. The characteristics of such an approach include: (i) An incremental or local learning approach is preferred where data is analyzed by the model sequentially, (ii) Only one data can be seen at each time by the model after which the model is updated, (iii) The data is discarded after updating the model, (iv) No prior-knowledge about the data is required. Nguyen, (2012) categorized the NFMs in 3 classes as described in the following:

Class I: The NFMs of this class cannot learn in incremental manner but they adopt a batch or pseudo-incremental learning schema. This NFMs can be categorized as static (non-constructive) or self-organizing (constructive). In a static NFM like as ANFIS, the structure of the model is fixed and does not change. Also the learning scheme in this model is batch-learning. In the self-organizing model of Class I, the model structure can change but the learning schema is pseudo-incremental. In this manner a copy of training data is retained for the tuning. SOFNN (Leng et al., 2004), DFNN (Wu and Er, 2000) are examples of this models.

Class II: The NFMs can learn in an incremental manner but limited to time-invariant environments where the underlying data-generation process does not change. These models have a self-organizing structure and DENFIS (Kasabov and Song, 2002) and FITSK (Quah and Quek, 2006) are examples of these models. In these models, prior knowledge of the upper/lower bound of the data and/or the number of clusters must be known in advance.

Class III: The NFMs which can learn incrementally without need for prior data. These models are able to handle the time-variant data and are self-evolving. SONFIN (Juang and Lin, 1998), FLEXFIS (Lughofer and Klement, 2005), eTS (Angelov and Filev, 2004) and TSK-FCMAC (Ting and Quek, 2009) are some examples of these type of models.

Although the Class III seems to be most suitable model for dealing with very complex problems, some disadvantages remain. Firstly, the model rule-base can only increase, secondly, interpretability is not considered in these models and lastly, the models cannot be applied in cases where the data exhibits regime shift/drift. A

self-evolving model namely Generic Self-Evolving Takagi-Sugeno-Kang or GSETSK address these issues. In the next 3 sections mainly the NFM architecture, clustering and parameter learning regarding to Class I NFM e.g. ANFIS and Class II NFM e.g. DENFIS is presented. Then in Section 3.5 the main NFM used in this thesis, GSETSK is detailed.

3.4.1 Neuro-Fuzzy Model Architecture

The general architecture applicable to NFMs applied in this thesis (ANFIS, DENFIS, and GSETSK) is presented in this section. As the main focus of this thesis is on GSETSK, only common information of the NFM architecture is presented here. Detailed formulation is presented for GSETSK architecture in Section 3.5.1. The general architecture of an NFM is shown in Figure 3.10, consisting of 6 layers:

Layer I or Input Layer: This layer signals the received crisps inputs to the Layer II.

Layer II or Fuzzification Layer: The fuzzification of the crisp values is done in this layer. DENFIS and ANFIS generally use Gaussian and bell-shaped membership functions, respectively.

Layer III or Rule Layer: The *firing strength* of each rule is computed in every node of this layer. This is done by using the AND (min) operator (Π operator in Figure 3.10).

Layer IV or Normalization Layer: The *normalized firing strength* is computed in this layer (N operator in Figure 3.10):

Layer V or Consequent Layer: The contribution of each rule in the NFM final result is reflected in each consequent node in the Consequent Layer. In this way the level of activation of input space (reflected in normalized firing strength) is multiplied by the respective linear output.

Layer VI or Summation Layer: This layer sums up the result of each rule to provide the final output of the model (Σ operator in Figure 3.10):

This general 6-layer structure and their functions are common between ANFIS, DENFIS and GSETSK. However, the ANFIS model is essentially a global model and has a fixed structure. Both DENFIS and GSETSK have a local learning approach which enables these models to have a dynamic structure. Table 3.1 highlights the differences which are essential to this chapter, between the three models. These differences are elaborated in the following sections.

Figure 3.10: General architecture of NFMs; in a static NFM, model structure is fixed (full line). In a dynamic NFM, the model structure can evolve during training as indicated by the dashed lines

Table 3.1: Comparison of NFMs.

NFM	Architecture	Learning Scheme	Clustering Method	Parameter Learning	
				Antecedent part	Consequent part
ANFIS	Fixed	Global	SC	Gradient Descent	LSE
DENFIS	Dynamic (expand only)	Local (Incremental with knowledge of min/max of data range required)	ECM	-	Weighted Recursive LSE
GSETSK	Dynamic (expand and contract)	Local (Incremental without need for knowledge of min/max of data range required)	MSGC	-	Localized Recursive LSE

3.4.2 Clustering

3.4.2.1 Global approach

Local and global models differ in the methodologies used to partition the input space. In a global model, the entire training dataset is used whereas for a local model, an incremental partitioning method is adopted. Common global clustering approaches adopted in ANFIS are grid partitioning and subtractive clustering or SC (Chiu, 1994), of which the latter is probably more well-known. An example of the SC method used to partition a unit 2-D input space based with batch training is shown in Figure 3.11(a). The procedure includes the following steps: (i) The first cluster center is chosen as the data point which has the highest potential defined by the d (Figure 3.11(b)):

$$d_i = \sum_{j=1}^n \exp \left\{ - \frac{\|x_i - x_j\|^2}{(r_a / 2)^2} \right\}, \quad (3.9)$$

where $j = (1, 2, \dots, n)$ is the index for the points located at predefined radius (r_a) for point x_i . (ii) All of the other points located at specific radius of the first cluster center are removed. Then the potential of the remaining points are evaluated to find the next cluster. (iii) This process iterates until all data points are included within the clusters formed. In SC a cluster is defined as a dense region of points which is separated by low-density regions from other clusters.

Figure 3.11: (a) SC method applied for 2-D input space, (b) Density measure for an arbitrary data point effective at radius r_a , where d is the Euclidean distance

3.4.2.2 Local Approach

The Evolving Clustering Method or ECM (Kasabov and Son, 2002) is a local clustering method used in DENFIS. This method incrementally tracks changes in the input space. The ECM (Figure 3.12) can be summarized by the following steps: (i) The location of the first data point in the unit hyperbox is adopted as the first cluster center with zero cluster radius. (ii) As the radius of the first cluster is zero

and hence the 2nd data point is not covered by the first cluster, the following action happens. With the arrival of the 2nd data point, the first cluster center is updated to a new position if the 2nd data point is within a pre-defined range of the 1st cluster center. For this purpose a circle is created with the most recent data location and its farthest point in the old cluster circle. Then the center and radius of the newly created circle are chosen as the center and radius of the updated cluster. If the 2nd data point is outside the pre-defined range, a new cluster center is created. This process of cluster adaptation and/or creation continues as each data point is presented one at a time. In ECM, the normalized Euclidean distance is used to define the distance between a data point $\mathbf{x}$ and an existing cluster center $\mathbf{y}$:

$$\|\mathbf{x} - \mathbf{y}\| = \left(\sum_{i=1}^q |x_i - y_i|^2 \right)^{1/2} / q^{1/2}, \mathbf{x}, \mathbf{y} \in \mathbf{R}^q \quad (3.10)$$

Figure 3.12: ECM Clustering mechanism for a newly arrived data point in 2-D input space

3.4.3 Parameter Learning

After partitioning of the input space by the specific clustering method, the output space should be optimized. This is done by introducing the concurrent training input-output pairs $(\mathbf{X}(t), d(t))$ to the model. The training is based on a batch of data in a global model while in a local model; training is based on the last seen input-output pair. Global NFMs such as ANFIS use the method of least square estimation

(LSE) to optimize the consequent parameters. The results of a global model is therefore more biased towards the average behavior of the system while DENFIS uses the weighted recursive LSE (Kasabov and Song, 2002) method and GSETSK adopts a localized version of recursive LSE (detailed in Section 3.5.3) for parameter learning.

3.5 GSETSK

The Generic Self-Evolving Takagi-Sugeno-Kang or GSETSK (Nguyen, 2012; Nguyen et al., 2015) is proposed for this study. The self-evolving property of this model makes it suitable as a fully-online model which can work with data streams without any prior assumption of the data distribution. GSETSK is an incremental model and is able to dynamically change its structure and parameters to account for the most data in the data stream. With every data arrival the following steps are performed in GSETSK:

1. First phase of structure learning involves updating the knowledge-base (MFs and rule-base) on the most recent input vector received. It is accomplished by using the Multidimensional-Scaling Growing Clustering method or MSGC which does not need the prior knowledge of data. In addition, the rule-merger algorithm in MSGC merges similar GMFs. As a result, a compact and distinguishable knowledge-base is generated using MSGC.
2. Reasoning (forecasting/modeling) with the most recent input.
3. Second phase of structural learning involves pruning obsolete rules. This property of GSETSK helps it to maintain a more compact and up-to-date rule-base since unwanted rules are discarded.
4. Updating the consequent parameters of the model based on the most recent observed output.

3.5.1 GSETSK Structure

GSETSK is a six-layer feedforward neural network structure with connections from Layer VI to Layer III via Layer V (Figure 3.13). The GSETSK layers and their functions are defined as follows:

Layer I (*input layer*): This layer consists of a number of *input nodes* (l_i) which are equal to the input dimensions ($i = 1, \dots, n$). The function of this layer is to make fuzzy singletons from each input dimension. Eq. (3.11) shows the output of Layer I for dimension i (z_i^I) which is equal to the value of that input dimension (x_i).

$$z_i^I = x_i, \quad i = 1, \dots, n \quad (3.11)$$

Layer II (*input linguistic layer*): This layer consists of some *input linguistic nodes* (l_{i,j_i}) which describe the firing labels in the antecedent part of TSK rules. The inputs are matched against *Gaussian membership functions* (GMFs) using Eq. (3.12).

$$z_{i,j_i}^{II} = \exp(-(x_i - c_{i,j_i})^2 / 2\sigma_{i,j_i}^2), \quad i = 1, \dots, n, \quad j_i = 1, \dots, J_i(t) \quad (3.12)$$

where z_{i,j_i}^{II} , c_{i,j_i} , and σ_{i,j_i}^2 are the output of Layer II, center of the GMF and variance of GMF respectively.

Layer III (*rule layer*): This layer consists of *rule nodes* (r_k). Using the AND (min) operation (Eq. (3.13)), a rule node computes the *firing strength of rule k* (r_k) by evaluating all fuzzified inputs connected to that rule ($z_{i,j_i}^{II,k}$).

$$r_k = \min(z_{i,j_i}^{II,k}), \quad k = 1, \dots, K(t) \quad (3.13)$$

Layer IV (*normalization layer*): This layer consists of the *normalization nodes* (N_k). The function of this layer is to compute the *average firing strength* of each rule

(ω_k) based on the firing strength of that rule (r_k) over the summation of the other rules firing strengths $(\sum_{k=1}^{K(t)} r_k)$.

$$\omega_k = r_k / \sum_{k=1}^{K(t)} r_k, \quad k = 1, \dots, K(t) \quad (3.14)$$

Layer V (*consequent layer*): The type of fuzzy inference system (FIS) used in GSETSK is based on the Takagi-Sugeno-Kang (TSK) (Takagi and Sugeno, 1985) rule model. Using a first-order TSK, a linear relationship is computed between the output $f_k(X)$ and input vector (X) as shown in Eq. (3.15). This is associated with the part of input space activated.

$$f_k(X) = b_{0k} + b_{1k}x_1 + \dots + b_{ik}x_i + \dots + b_{nk}x_n, \quad k = 1, \dots, K(t) \quad (3.15)$$

where $[b_{0k}, b_{1k}, \dots, b_{ik}, \dots, b_{nk}]$, are the parameters of the linear model which would be tuned during the parameter learning phase (Section 3.5.3).

Layer V consists of the *consequent nodes*. Output of each node (z_k^V) is the multiplication of output of the linear equation $(f_k(X))$ and the normalized firing strength fed to that node (ω_k) as shown in Eq. (3.16).

$$z_k^V = \omega_k f_k(X), \quad k = 1, \dots, K(t) \quad (3.16)$$

Layer VI (*summation layer*): The output of Layer VI (y) is the aggregation of the weighted output of the fired rules (z_k^V) as shown in Eq. (3.17).

$$y = \sum_{k=1}^{K(t)} z_k^V = \sum_{k=1}^{K(t)} \omega_k f_k(X), \quad (3.17)$$

3.5.2 GSETSK Structure Learning

The GSETSK architecture defined in Section 3.5.1 does not follow a fixed structure. GSETSK is initialized as an empty structure and grows as data is fed to the model sequentially. Based on the complexity of the problem, GMFs and rules can be created or removed. There are two main mechanisms within GSETSK that allow for this dynamism. First is the Multidimensional-Scaling Growing Clustering method or MSGC, and second is Rule Pruning.

Figure 3.13: GSETSK architecture

3.5.2.1 Multidimensional-Scaling Growing Clustering (MSGC)

Although ECM (discussed in Section 3.4.2.2) allows for new cluster centres to be incrementally created, it does not have the capability of removing cluster centres that may have become obsolete. Multidimensional-Scaling Growing Clustering (MSGC) is an online clustering method used in GSETSK. The main distinction

between MSGC and SC or ECM is that both SC and ECM work on a unit hyperbox, which is created by data normalization. This means that prior data of min/max values is required. In MSGC, data are projected onto every single input space with the arrival of each data point. The data are thus evaluated in a 1-D space rather than multi-dimensional space, and thus precludes the need for data normalization. The MSGC clustering method ensures that the NFM maintains a non-overlapping knowledge-base. MSGC also includes a merging mechanism that manages the knowledge-base making it compact. As discussed earlier, there is no rule or network at the beginning. When the first input data vector arrives, it is distributed to different 1-dimensional spaces. In this way there would be several GMFs for different dimensions in which the center of each GMF ($c_{i,1}$) is given the input value for that space (x_i). Also the width for each GMF ($\sigma_{i,1}$) is a predefined constant (v_i) which can be set to an arbitrary value or based on expert knowledge. Large values are recommended for this parameter to ensure a smaller number of GMFs in the entire universe of discourse of each input space.

$$c_{i,1} = x_i, \quad \sigma_{i,1} = v_i \quad (3.18)$$

From the presentation of second data onwards, GSETSK is run partially up to Layer III and the firing strength ($r_k(X(t))$) is computed for those rules which cover that input data vector as shown in Eq. (3.13). Then the maximum firing strength (ρ) is determined using Eq. (3.19):

$$\rho = \arg \max_{1 \leq k \leq K(t)} r_k(x(t)) \quad (3.19)$$

If the maximum strength is less than the firing threshold ($\theta \in (0,1)$), a new rule will be created. The firing threshold controls the number of rules to be created. Higher values of this parameter means more rules to be created but comprehensibility would be compromised. In this study θ is predefined as 0.6 in order to achieve the balance between accuracy and comprehensibility/interpretability. To create a new

rule, firstly the closest GMF (ε) to the current input data vector is determined using Eq. (3.20):

$$\varepsilon = \arg \max_{1 \leq j \leq J_i(t)} \exp(-(x_i(t) - C_{i,j})^2 / 2\sigma_{i,j}^2) \quad (3.20)$$

In this equation $C_{i,j}$ and $\sigma_{i,j}^2$ are the center and variance of the various GMFs to be evaluated. $x_i(t)$ is the current input data of a specific dimension. When a new GMF is created, the centroid of the new GMF would be set as the input data value. Eq. (3.21) is used to determine the width of the new GMF ($\sigma_{i,Ji(t)}$) where $x_i(t)$ and $C_{i,\varepsilon}$ are the input data and the centroid of the closest GMF respectively. The predefined constant η describing the *degree of overlap* between two GMFs is set at 0.5 (Nguyen et al., 2015) in this study. The lower value of this parameter means lesser overlap between the newly created GMF and it's nearest GMF.

$$\sigma_{i,Ji(t)} = \eta \left| x_i(t) - C_{i,\varepsilon} \right| \quad (3.21)$$

MSGC knowledge-base management is based on the concept of *similarity*, which determines if the newly created GMFs are to be kept, discarded or merged with the closest one. A fuzzy subset-hood measure (Yen and Langari, 1999) is used (see Eq. (3.22)) for this purpose to determine the similarity between two GMFs ($\mu_A(X)$) and ($\mu_B(X)$):

$$S(A, B) = \frac{\max_{X \in \cup} (\min(\mu_A(X), \mu_B(X)))}{\max_{X \in \cup} (\mu_A(X))} = \frac{\max_{X \in \cup} (\min(\mu_A(X), \mu_B(X)))}{1} \quad (3.22)$$

By denoting the newly created GMF as $\mu_A^{(c_{i,J_i}, \sigma_{i,J_i})}$ and closest GMF as $\mu_B^{(c_{i,\varepsilon}, \sigma_{i,\varepsilon})}$, and based on their similarity (S) the following actions can be adopted (Figure 3.14) :

- (i) If $S > \text{thresA}$, then the newly created GMF would be replaced by the closest one.

(ii) If $\text{thresA} > S > \text{thresB}$, then the two GMFs are merged.

(iii) If $S < \text{thresB}$, then the newly created GMF is accepted.

The two predefined similarity thresholds thresA and thresB control the number of rules to be created. Higher values of these parameters mean more GMFs can be created. Reasonable values used in this study are 0.8 and 0.7 for thresA and thresB , respectively, as suggested by (Turksen and Zhong, 1988). In the case of GMFs merging, Eq. (3.23) and Eq. (3.24) are used to determine centroid and variance of the newly merged GMF.

$$C_{new} = (C_1 + C_2 + (\sigma_1 - \sigma_2)\sqrt{\pi}) / 2 \quad (3.23)$$

$$\sigma_{new} = (C_1 - C_2 + (\sigma_1 + \sigma_2)\sqrt{\pi}) / (2\sqrt{\pi}) \quad (3.24)$$

where C_1 and C_2 are the centers and σ_1 and σ_2 are standard deviations of two GMFs to be merged. Overlapping GMFs are merged in order to control the size of new GMF via the computation of a *willingness parameter* or $WP(t)$ (see Eq. (3.25)). Two GMFs will not be allowed to be merged if $WP(t) \leq 0$.

$$WP(t) = WP(t_u) - (1.5 - WP(t_u)) \times (1 - S) \quad (3.25)$$

where $WP(t_u)$ is the previous willingness parameter. The initial willingness potential $WP(0)$ was set to 0.5 (Nguyen et al., 2015).

Figure 3.14: GSETSK knowledge-base management

3.5.2.2 Rule Pruning

Rule pruning allows GSETSK to discard obsolete rules and maintain the most relevant rules. This is supported by the backward connections from Layer VI to Layer III via Layer V (please refer to Figure 3.13). In this way the *backward firing strength* (Eq.(3.27)) is computed and together with the *forward firing strength* (Eq.(3.13)) can be used to evaluate the potential of a rule to be kept or pruned. This is based on the Hebbian learning mechanism (Whitlock et al., 2006).

As stated in Eq. (3.26), at time t , the actual output $d(t)$ and the simulated output for each rule ($f_k(X(t))$, Eq. (3.15)) are used to compute the error ($e_k(t)$) for each rule:

$$e_k(t) = \left| d(t) - f_k(X(t)) \right|, \quad k = 1, \dots, K(t) \quad (3.26)$$

These errors are used to form a GMF with zero mean and variance $\sigma_{back}(t)$ given by Eq. (3.27).

$$\sigma_{back}(t) = \frac{\sum_{k=1}^{K(t)} e_k(t)}{K(t)\sqrt{\pi}} \quad (3.27)$$

This GMF shows how close the computed value ($f_k(X(t))$) is to the desired ($d(t)$). The backward firing strength ($r_k^{back}(t)$) is then computed using Eq. (3.28) in order that rules with higher value have a closer output to the desired one.

$$r_k^{back}(t) = \exp((-e_k(t)^2) / (2\sigma_{back}(t)^2)) \quad (3.28)$$

The fuzzy rule potential $P_k(t)$ is computed using Eq. (3.29) in which $P_k(t-1)$ is the previous fuzzy rule potential. The initial fuzzy potential for every rule is set to 1 when it is newly created.

$$P_k(t) = \gamma P_k(t-1) + r_k(x(t)) \times r_k^{back}(d(t)), \quad k = 1, \dots, K(t) \quad (3.29)$$

In Eq. (3.29), $\gamma = [0.97, 0.99]$ (Tung and Quek, 2010) is the *rule pruning forgetting factor*. Smaller value of this parameter means a faster pruning of rules. This parameter is set to 0.97 in this study. The computed fuzzy rule potential is contrasted against the threshold, thresP . The rule would be pruned if $P_k(t) < \text{thresP}$. The threshold thresP is a predefined value which is set at 0.5 (Nguyen et al., 2015). This implies that a rule will be pruned when it losses half of its initial potential which was initially set as 1.

3.5.3 GSETSK Parameter Learning

After updating the structure of GSETSK (Section 3.5.2.1) and reasoning (testing) based on the most recent data, the model parameters would be updated. As discussed earlier in Section 3.5.1, GSETSK uses the 1st order TSK which utilizes a linear equation for the consequent part of each rule (Eq. (3.15)). By substituting this equation to Eq. (3.16), the final output can be computed using Eq. (3.30):

$$y = \sum_{k=1}^{k(t)} \omega_k f_k(X) = \sum_{k=1}^{k(t)} \omega_k [b_{ok} + b_{1k}x_1 + \dots + b_{ik}x_i + \dots + b_{nk}x_n], \quad (3.30)$$

where ω_k is the normalized firing strength of rule k , and

$B_k = [b_{0k}, \dots, b_{1k}, \dots, b_{nk}]^T$ is the consequent parameter vector of rule k to be tuned.

GSETSK applies a *localized version of recursive linear-least-square* (RLS) (Quah and Quek, 2006) for this purpose.

To understand how this vector is tuned, only one rule R_k with two inputs (x_1, x_2) in GSETSK is used to demonstrate the tuning process. Further, the rule R_k remains in GSETSK while it is presented with T number of input-output pairs $(X(1), d(1)), \dots, (X(t), d(t)), \dots, (X(T), d(T))$. This can be presented in a matrix form as in Eq. (3.31), (3.32).

$$\begin{bmatrix} \omega_k(1) & \omega_k(1)x_1(1) & \omega_k(1)x_2(1) \\ \vdots & \vdots & \vdots \\ \omega_k(t) & \omega_k(t)x_1(t) & \omega_k(t)x_2(t) \\ \vdots & \vdots & \vdots \\ \omega_k(T) & \omega_k(T)x_1(T) & \omega_k(T)x_2(T) \end{bmatrix} \times \begin{bmatrix} b_{0k} \\ b_{1k} \\ b_{2k} \end{bmatrix} = \begin{bmatrix} \omega_k(1)d(1) \\ \vdots \\ \omega_k(t)d(t) \\ \vdots \\ \omega_k(T)d(T) \end{bmatrix} \quad (3.31)$$

or

$$A \times B = D \quad (3.32)$$

$B = \begin{bmatrix} b_{0k} & b_{1k} & b_{2k} \end{bmatrix}^T$ can be iteratively estimated using RLS (Hayes, 1996) as shown in Eq. (3.33).

$$\begin{cases} B_{p+1} = B_p + C_p + a_{p+1}^T (d - a_{p+1} B_p) \\ C_{p+1} = \frac{1}{\lambda} \left[C_p - \frac{C_p a_{p+1}^T a_{p+1} C_p}{\lambda + a_{p+1} C_p a_{p+1}^T} \right] \end{cases} \quad (3.33)$$

where a_p is the p^{th} row of the matrix A . $\lambda = (0,1]$ is the *learning forgetting factor* for parameter learning which is set at 0.99 in GSETSK. For initialization, $C_0 = gI$

where ϑ is a positive large number and I is the identity matrix. Also B_0 is a zero matrix.

For the newly created rule $R_{k(t+1)}$, consequent parameters are the weighted average of the parameters of the other rules (Angelov and Filev, 2004) which can be computed using Eq. (3.34).

$$b_{i,K(t+1)} = \sum_{k=1}^{k(t)} \omega_k b_{ik}, \quad i = 1, \dots, n \quad (3.34)$$

where $[b_{ok}, \dots, b_{ik}, \dots, b_{nk}]^T$ and ω_k are the parameter vector of the rule and the normalized firing strength of rule R_k respectively.

Figure 3.15 shows the flowchart of GSETSK mechanism described in this section.

Figure 3.15: GSETSK flowchart

3.6 Study Area and Data

Two datasets from different geographical locations were used in this study. One is from Klippan_2 Catchment in Sweden, which was used for rainfall-runoff forecasting and the other is Mekong River which was used for river routing forecasting.

3.6.1 Klippan_2 Catchment

Klippan_2 Catchment has an area of 242.9 km^2 and is located in Southern Sweden (Figure 3.16). Klippan_2 is a sub-basin of Rönne River Catchment which drains an area of 1900 km^2 into the Baltic Sea. It is a rural catchment which mainly consists of woodlands and the soil is made up of moraine, peat and coarse material. River discharge (Q), precipitation (P) and temperature (T) are the variables used for forecasting the runoff flow in this catchment. Temperature was taken as a surrogate parameter for snowmelt. This data is provided on a daily basis by the Swedish Meteorological and Hydrological Institute (SMHI, 2013).

3.6.2 Lower Mekong River

The Mekong River is the 10 largest river in the world. With a length of about 4,880 km, it annually drains $470 \times 10^9 \text{ m}^3$ water from an area of $790,000 \text{ km}^2$ (Gupta, 2007). The Mekong River Catchment extends from eastern watershed of Tibetan Plateau to the Mekong Delta in Cambodia and Vietnam (Figure 3.17). The main river passes through China, Myanmar, Laos, Thailand, Cambodia and Vietnam discharging into the sea. The Mekong River is divided to two parts the Upper Mekong River which has a length about 2,200 km which is mainly in China and extends to Golden Triangle where China, Laos, Myanmar, Thailand borders meet each other. The Lower Mekong starts from this point to the river mouth in Vietnam and has a length of 2,700 km. The Lower Mekong includes a watershed area of $600,000 \text{ km}^2$ in Laos, Thailand, Cambodia and Vietnam. The elevation ranges from 2,806 m in the north to sea level at Mekong Delta in Vietnam (Figure 3.18). The

wet season extends from June to October. Tropical typhoons mostly occur during September and October from South China Sea to the east and southeast of the Mekong Basin and extreme water levels and flooding may happen during this time. Hydrometric gauge stations are installed along the Mekong River for measuring the river stage. Water level data from three stations, Thakhek (TK), Savannakhet (SV) and Pakse (PK) (Figure 3.18) were used in this study.

Figure 3.16: Klippan_2 Catchment (adopted from Talei (2013))

Figure 3.17: Mekong River watershed (adopted from Mekong River Commission (2011))

Figure 3.18: Lower Mekong River watershed (adopted from Mekong River Commission (2011))

3.7 Evaluation Indices

There are two different sets of evaluation indices used in this study.

3.7.1 Goodness of Fit Indices

Goodness of fit indices, are useful tools for evaluating the entire dataset. These indices are used for evaluation of the overall model performance in this study. In addition, they were used for choosing the best model output-input scenario in cross validation analysis. These include root mean square error (RMSE), mean absolute error (MAE), coefficient of efficiency (CE), and coefficient of determination (R^2), as presented in Eq. (3.35) to Eq. (3.38), respectively. In these equations, O_i and S_i are the observed and forecasted values for instance i ($i = 1, 2, \dots, n$), and $\bar{O}$ and $\bar{S}$ are the average of the observations and the average of simulations, respectively.

$$RMSE = \sqrt{\frac{\sum_{i=1}^n (O_i - S_i)^2}{n}} \quad (3.35)$$

$$MAE = \frac{1}{n} \sum_{i=1}^n |O_i - S_i| \quad (3.36)$$

$$CE = 1 - \frac{\sum_{i=1}^n (O_i - S_i)^2}{\sum_{i=1}^n (O_i - \bar{O})^2} \quad (3.37)$$

$$R^2 = \left[\frac{\sum_{i=1}^n (O_i - \bar{O})(S_i - \bar{S})}{\sqrt{\sum_{i=1}^n (O_i - \bar{O})^2} \times \sqrt{\sum_{i=1}^n (S_i - \bar{S})^2}} \right]^2 \quad (3.38)$$

3.7.2 Threshold-based Indices

To have a good evaluation of the performance of the model specifically on the floods, threshold-based indices were used. These indices give a better understanding of the capability of the model to be used in a FFS. They include critical success index (CSI), probability of detection (POD) and false alarm ratio (FAR) as defined in Eq. (3.39) to Eq. (3.41) respectively.

$$CSI = TA / (TA + FA + FR) \quad (3.39)$$

$$POD = TA / (TA + FR) \quad (3.40)$$

$$FAR = FA / (TA + FA) \quad (3.41)$$

Where TA (true acceptance) is number of data points greater than the threshold both in the observed and forecasted flows, FA (false acceptance) is the number of measured data less than the threshold but were forecasted to be greater than the threshold, and FR (false rejection) is the number of measured data greater than the threshold but were forecasted to be lesser than the threshold. For CSI and POD, values closer to 1 are desirable while FAR values closer to 0 are desirable.

3.8 Benchmarks

3.8.1 Unified Runoff Basin Simulation Model

The Unified Runoff Basin Simulation (URBS) (Carrol, 2007) was adopted as the benchmark for the river routing application. URBS is the operational flood forecasting model for Mekong River used by Regional Flood Management and Mitigation Center (RFMMC) of the Mekong River Commission (MRC). This model also has been used by Australian Bureau of Meteorology (BoM), and Changjiang Water Resources Commission (CWRC), China. URBS is a physically-based modeling approach which uses a network of conceptual storages to represent the stream networks and reservoirs. URBS consists of the two following modeling approaches. (i) Rainfall-runoff model which converts the gross rainfall to net or

excess rainfall, and (ii) Runoff routing model which converts the excess rainfall to flow. There are two runoff routing models namely URBS Basic and Split routing model. In URBS Basic runoff routing model, a single non-linear reservoir is represented by lumping the catchment and channel storage of each sub-catchment together. Then each reservoir is represented by storage-discharge relation (Eq. (3.42)).

$$S = k_c Q^m = \frac{\alpha f L n (1 + F)^2}{\sqrt{S_c} (1 + U)^2} Q^m \quad (3.42)$$

where S is the catchment and channel storage, Q is the discharge, K_c is a non-linear routing constant, m is a catchment non-linearity parameter, α is a lag parameter, f is a reach length factor, L is the reach length, n is channel roughness, F is fraction of forested sub-catchment, S_c is channel slope, and U is fraction of urbanized sub-catchment. In URBS Split runoff routing model, catchment and channel routing of each sub-catchment are separated. In this model the travel time from the perimeter of the catchment to its center (T) is calculated as in Eq. (3.43):

$$T = \sqrt{\frac{A}{\pi}} / v \quad (3.43)$$

where A and v are the area of the sub-catchment and the flow velocity, respectively. The runoff is routed through the catchment by Eq. (3.44):

$$S_{catchment} = \frac{\beta \sqrt{A} (1 + F)^2}{(1 + U)^2} Q^m \quad (3.44)$$

where $S_{catchment}$ and β are the catchment storage and lag parameter respectively and the non-linear Muskingum model (Eq. (3.45)) is taken as the flow routing model:

$$S_{channel} = \alpha f \frac{nL}{\sqrt{S_c}} [x Q_u + (1 - x) Q_d]^P \quad (3.45)$$

where $S_{channel}$ is channel storage, α is the channel lag parameter, Q_u is inflow at upstream end of reach, Q_d is the outflow at downstream end of channel reach, x is

the Muskingum translation parameter, and p is the Muskingum non-linearity parameter. More details can be found in (Carrol, 2007). The results for the URBS model at Pakse station for the monsoon seasons (June to October) of 2010 to 2011 were obtained from MRC (2015) and used as benchmark for this study.

3.8.2 HBV Model

Hydrologiska Byråns Vattenbalansavdelning or HBV (Bergstrom and Forsman, 1973) is a conceptual rainfall-runoff model used by Swedish Meteorological and Hydrological Institute (SMHI). It is based on the following water balance equation.

$$Q = P - E - \Delta(S^* + SM + UZ + LZ) \quad (3.46)$$

where Q , P , and E indicate water fluxes for discharge, precipitation and evaporation respectively. S^* and SM are snow and soil moisture water storage components. Also UZ and LZ are water storage component for upper zone and lower zone of groundwater respectively.

The results for the HBV model for 1 Jan 1980 to 31 Dec 1985 were obtained from Talei et al. (2013) and used as benchmark for this study.

CHAPTER 4: RULE BASE DYNAMICS OF NEURO-FUZZY MODELS FOR FLOW FORECASTS

4.1 Introduction

From the review of the usage of NFMs for flow/flood forecasting presented in Chapter 2, it can be summarized that almost exclusively; all studies used the global learning approach or batch training in ANFIS which had a fixed rule base. Unlike ANFIS, modern NFMs however, incorporate a flexible or dynamic model structure and hence rule base. For example DENFIS employ a local learning technique, allowing the model to respond to changes through a modification of the rule base in an incremental fashion. As detailed in Chapter 2, Talei et al. (2013) used DENFIS for runoff forecasting. Also Hong and White, (2009) introduced a dynamic neuro-fuzzy local modeling system (DNFLMS) which is an online TSK NFM. They used DNFLMS to forecast the flow in Waikoropupu Springs, Takaka Valley, New Zealand. One difficulty that has arisen recently in adopting local learning in NFM is although the model is able to adapt its rule base, which is done by adding new rules; however, the reverse is not possible, i.e. rules that are outdated are not removed. This means that as the model evolves with time, the rule base expands and the model may become too complex leading to deterioration in interpretability and/or performance. In addition, current NFMs such as DENFIS require information on the upper and lower bound of historical data. This is not always available, or even if it is available, the model will not be applicable if the test data range exceeds the historical upper/lower bounds.

Recently, models such as the GSETSK have been reported in the computer science and engineering literature. GSETSK uses the MSGC clustering method and the incorporation of Hebbian-based rule pruning algorithm (Whitlock et al., 2006) with MSGC results in a dynamic NFM which can expand or shrink its rule base. In addition, MSGC does not need any prior knowledge of data which may be vital for applications where data is limited. Given these developments in NFMs including the introduction of local learning, clustering and pruning mechanisms, there is now

a need to critically assess these developments for hydrologic forecast applications. In this chapter, the objectives are (i) to assess local versus global learning strategies in NFM, and (ii) study the importance of rule base dynamics in relation to hydrologic forecasts. These objectives will be studied in the context of rainfall-runoff and flow routing applications and results are compared against physically based models used as a baseline. The rainfall-runoff application used precipitation, runoff and temperature data for Klippan_2 Basin in Sweden and the river routing application used river stage data from three gauging stations in the Lower Mekong River.

4.2 Methodology

4.2.1 Study Area and Data

Two datasets were used in this study. The first dataset belongs to Klippan_2 Basin which was used for the rainfall-runoff study (more detailed in Section 3.6.1). The main variables used for runoff forecasting in this basin are river discharge (Q), precipitation (P) and temperature (T). Precipitation includes both rainfall and snow, and temperature was used as a surrogate for snowmelt. Thirty years of daily data available from the Swedish Meteorological and Hydrological Institute website (SMHI, 2013) was used. This contains 10,947 data samples, ranging of 11 Jan 1961 to 31 Dec 1990. The second dataset from the Lower Mekong River (more detailed in Section 3.6.2) was used as a river routing application. The water level at two upstream gauging stations Thakhek (TK) and Savannakhet (SV) were used to predict the stage at Pakse (PK) in this study. Thakhek station is approximately 110 km upstream of Savannakhet which is in turn 240 km upstream of Pakse. The data used in this study consists of 9 years (1994-1999 and 2009-2011) of data obtained for the flood season which occurs from June to October of each year (MRC, 2015a).

4.2.2 Model Inputs

As discussed earlier in Section 2.4, there are two kinds of inputs which can be excluded during input selection, (i) irrelevant inputs, (ii) correlated inputs. As discussed in Chapter 2, all of the applied input selection methods only exclude irrelevant inputs. This is while excluding the correlated inputs and hence dimensionality reduction is recommended. In this chapter a preprocessing procedure was conducted to determine the model inputs via a two-step procedure: (i) select the independent variables for dimensionality reduction, and (ii) adopting a 10-fold cross validation method (Kohavi, 1995) to ascertain the best combinations of the independent inputs to be used for the models. The Monte Carlo Evaluative Selection (MCES) method (Quah and Quek, 2007) was adopted as the method to select the relevant and independent inputs. In this method, irrelevant and correlated inputs are identified based on weight ranking. Input variables which are redundant (irrelevant and correlated) are assigned negative weights while the recommended inputs are weighted positively. The weights are generated after an evaluative feedback procedure is collected using an induction method. Details of MCES can be found in Quah and Quek (2007). For the Klippan_2 dataset, the 1 day ahead forecast of discharge, Q_1 , is based on a selection of the following possible inputs:

$$Q_1 = f(Q_i, P_i, T_i), \quad i = -9, -8, \dots, -1, 0 \quad (4.1)$$

where the subscript i refers to time (days), and $i = 0$ denotes the current time. For the routing application for the Lower Mekong River, the water level at Pakse Station 1, 3 and 5 days ahead, PK_j , is estimated based on:

$$PK_j = f(TK_i, SV_i, PK_i), \quad j = 1, 3, 5 \quad i = -7, -6, \dots, -1, 0 \quad (4.2)$$

where $j = 1, 3, 5$ days is the forecast lead time and $i =$ time (day), and $i = 0$ denotes the current time.

In the second step of the input selection procedure, combinations of the independent inputs selected by MCES were tested and for each combination, a 10 fold cross validation evaluation method was adopted to identify the best input-output combination set to be eventually adopted by the model. The cross validation

procedure first divides the entire dataset into 10 equally-sized parts or folds, with nine folds adopted as the training dataset and the remaining fold for testing. This procedure was repeated ten times, each time choosing a new training/testing dataset combination. Error statistics were computed for each of 10 testing dataset, and then they were averaged. Finally the input combination that had the least error was selected to be the inputs to the model.

4.2.3 Procedure

GSETSK, DENFIS, and ANFIS were used for rainfall-runoff modeling with data from a Swedish Catchment and for flow routing modeling for the Lower Mekong River, respectively. The NFMs were run in the following manner:

(i) ANFIS is a global model and requires separate training and testing datasets (Table 4.1). The training data was used for rule base generation and optimization in ANFIS before being used on the testing data.

(ii) As GSETSK is an online model, the model does not require separate training and testing datasets (Table 4.1). The 1st input-output data sample was for model initialization and predictions started from the 2nd data point onwards. During the prediction stage, GSETSK first makes a forecast, after which the forecast value is compared against the measured data, once it becomes available. Based on a comparison between the predicted and actual values, the model is then updated. This process continues for the entire dataset with the arrival of each data point.

(iii) The incremental version of the DENFIS was applied which allowed the model to be run as an online model, similar to GSETSK. The limitation of DENFIS however is that the model requires training data for normalization and initialization the rule base. Once the rule base is initialized, the dynamic feature of DENFIS enables more rules to be added to the rule base during the model run. The period of data used in the training and testing phases of the DENFIS model are shown in Table 4.1.

The results from the NFMs were compared against results obtained from the physically-based models HBV (detailed in Section 3.8.1) and URBS (detailed in Section 3.8.2), adopted as benchmarks, for the Swedish catchment and the Lower Mekong River, respectively. The HBV model (Swedish dataset) results are available from 1 Jan 1980 to 31 Dec 1985 and the URBS model (Lower Mekong River) results are available for the monsoon seasons (June to October) of 2010 to 2011, for comparison.

Table 4.1: Data used by different NFMs.

NFM	Data Function	Rainfall-Runoff (Klippan_2)			River Routing (Mekong River)		
		From	To	No. of Data Samples	From	To	No. of Data Samples
ANFIS	Training	11-Jan-61	31-Dec-79	6,929	1994	1999	918
	Testing	1-Jan-80	31-Dec-90	4,018	2009	2011	413
DENFIS	Model Initialization	11-Jan-61	10-Jan-62	365	1994		153
	Continuous Testing/Training	11-Jan-62	31-Dec-90	10,582	1995-1999, 2009-2011		1,178
GSETSK	Model Initialization	11-Jan-61		1	1-Jun-1994		1
	Continuous Testing/Training	12-Jan-61	31-Dec-90	10,946	1994-1999, 2009-2011		1,330

4.3 Results and Discussion

4.3.1 Input Selection

For the rainfall-runoff application, nine variables returned positive weights (out of a total of 31 variables, Eq. (4.1)) when analyzed with MCES (Table 4.2). Based on this analysis six variables, Q_0 , Q_{-2} , Q_{-4} , P_0 , T_{-1} , P_{-1} , which had significant positive weightings were selected for cross validation analysis. A total of 16 (more obvious choices) combinations of these six variables are tested with the cross validation method and the results are shown in Table 4.3 which suggests that Q_0 , P_0 and T_{-1} (K_{11}) be chosen to forecast Q_1 by the NFMs in the rainfall-runoff forecast model:

$$Q_1 = f(Q_0, P_0, T_{-1}) \quad (4.3)$$

Similar analysis with MCES was carried out for the Lower Mekong data, and the results are shown in Table 4.4. Following MCES, cross validation was used to determine the best input-output combinations for different combinations (Table 4.5 shows the cross validation results). Following the results in Table 4.5, the 1, 3, and 5 step-ahead forecasts for Pakse can be represented by:

$$PK_1 = f(PK_0, SV_{-1}, SV_{-3}, TK_{-3}) \quad (4.4)$$

$$PK_3 = f(PK_{-2}, PK_{-3}, PK_{-4}, SV_{-3}, SV_{-5}) \quad (4.5)$$

$$PK_5 = f(PK_{-4}, PK_{-5}, PK_{-6}, SV_{-5}, SV_{-7}) \quad (4.6)$$

These results are consistent with the travel times for a flood wave between the stations. As the flow is estimated to take 1-2 days from Savannakhet to Pakse, and 2-3 days from Thakhek, to Pakse.

Table 4.2: MCES results for rainfall-runoff application (only positive values are shown).

Rank	1	2	3	4	5	6	7	8	9
Variable	Q_0	Q_{-2}	Q_{-4}	P_0	T_{-1}	P_{-1}	Q_{-6}	P_{-8}	P_{-6}
Weight	0.626	0.108	0.03	0.026	0.009	0.007	0.002	0.001	0.0002

Table 4.3: Cross validation results for rainfall-runoff application

Input combination	Inputs					CE	R ²	RMSE (mm/d)	MAE (mm/d)
K1	Q ₀	-	-	-	-	0.77	0.86	0.55	0.42
K2	Q ₀	Q ₋₂	-	-	-	0.76	0.84	0.57	0.44
K3	Q ₀	Q ₋₄	-	-	-	0.81	0.85	0.50	0.35
K4	Q ₀	P ₀	-	-	-	0.84	0.89	0.46	0.34
K5	Q ₀	T ₋₁	-	-	-	0.83	0.83	0.48	0.27
K6	Q ₀	Q ₋₂	Q ₋₄	-	-	0.77	0.84	0.55	0.41
K7	Q ₀	Q ₋₂	P ₀	-	-	0.82	0.89	0.49	0.36
K8	Q ₀	Q ₋₂	T ₋₁	-	-	0.83	0.84	0.47	0.27
K9	Q ₀	Q ₋₄	P ₀	-	-	0.85	0.89	0.45	0.31
K10	Q ₀	Q ₋₄	T ₋₁	-	-	0.80	0.82	0.50	0.30
K11	Q₀	P₀	T₋₁	-	-	0.89	0.90	0.38	0.20
K12	Q ₀	Q ₋₂	Q ₋₄	P ₀	-	0.85	0.89	0.44	0.30
K13	Q ₀	Q ₋₂	Q ₋₄	T ₋₁	-	0.83	0.84	0.47	0.27
K14	Q ₀	Q ₋₂	P ₀	T ₋₁	-	0.88	0.89	0.39	0.20
K15	Q ₀	Q ₋₄	P ₀	T ₋₁	-	0.89	0.89	0.39	0.20
K16	Q ₀	Q ₋₂	Q ₋₄	P ₀	T ₋₁	0.77	0.86	0.55	0.42

Table 4.4: MCES results for river routing application (only positive values are shown)

1 day ahead			3 days ahead			5 days ahead		
Rank	Variable	Weight	Rank	Variable	Weight	Rank	Variable	Weight
1	PK ₀	1.99	1	PK ₂	1.44	1	PK ₄	1.35
2	SV ₋₁	0.40	2	SV ₋₃	0.48	2	PK ₅	0.52
3	SV ₋₃	0.26	3	SV ₋₅	0.21	3	SV ₋₅	0.41
4	TK ₋₃	0.18	4	PK ₄	0.20	4	SV ₋₇	0.39
5	PK ₂	0.002	5	PK ₃	0.06	5	PK ₆	0.23

Table 4.5: Results of cross validation procedure for different model input scenarios for river routing forecasts.

Forecast Lead Time	Scenario	Inputs					RMSE (m)	MAE (m)
1 day	M ₁	PK ₀					0.29	0.21
	M ₂	PK ₀	SV ₋₁				0.52	0.41
	M ₃	PK ₀	SV ₋₃				0.66	0.49
	M ₄	PK ₀	TK ₋₃				0.68	0.53
	M ₅	PK ₀	PK ₋₂				0.43	0.30
	M ₆	PK ₀	SV ₋₁	SV ₋₃			0.40	0.30
	M ₇	PK ₀	SV ₋₁	TK ₋₃			0.40	0.31
	M ₈	PK ₀	SV ₋₁	PK ₋₂			0.43	0.32
	M₉	PK₀	SV₋₁	SV₋₃	TK₋₃		0.24	0.19
	M ₁₀	PK ₀	SV ₋₁	SV ₋₃	PK ₋₂		0.32	0.22
	M ₁₁	PK ₀	SV ₋₁	SV ₋₃	TK ₋₃	PK ₋₂	0.31	0.21
3 days	M ₁	PK ₋₂					0.72	0.54
	M ₂	PK ₋₂	SV ₋₃				0.84	0.61
	M ₃	PK ₋₂	SV ₋₅				1.11	0.83
	M ₄	PK ₋₂	PK ₋₄				1.06	0.81
	M ₅	PK ₋₂	PK ₋₃				0.90	0.68
	M ₆	PK ₋₂	SV ₋₃	SV ₋₅			0.42	0.32
	M ₇	PK ₋₂	SV ₋₃	PK ₋₄			0.48	0.36
	M ₈	PK ₋₂	SV ₋₃	PK ₋₃			0.26	0.18
	M ₉	PK ₋₂	SV ₋₃	SV ₋₅	PK ₋₄		0.33	0.23
	M ₁₀	PK ₋₂	SV ₋₃	SV ₋₅	PK ₋₃		0.2	0.13
	M₁₁	PK₋₂	SV₋₃	SV₋₅	PK₋₄	PK₋₃	0.12	0.08
5 days	M ₁	PK ₋₄					1.06	0.81
	M ₂	PK ₋₄	PK ₋₅				0.26	0.19
	M ₃	PK ₋₄	SV ₋₅				0.53	0.41
	M ₄	PK ₋₄	SV ₋₇				0.70	0.53
	M ₅	PK ₋₄	PK ₋₆				0.49	0.35
	M ₆	PK ₋₄	PK ₋₅	SV ₋₅			0.53	0.41
	M ₇	PK ₋₄	PK ₋₅	SV ₋₇			0.60	0.44
	M ₈	PK ₋₄	PK ₋₅	PK ₋₆			0.22	0.15
	M ₉	PK ₋₄	PK ₋₅	SV ₋₅	SV ₋₇		0.33	0.25
	M ₁₀	PK ₋₄	PK ₋₅	SV ₋₅	PK ₋₆		0.22	0.15
	M₁₁	PK₋₄	PK₋₅	SV₋₅	SV₋₇	PK₋₆	0.19	0.13

4.3.2 Comparison of Results

Figure 4.1 shows the results (computed over the part of the test period which coincides with HBV results) of forecasting with NFM models benchmarked against the physically-based HBV model obtained from (Talei et al., 2013) for the rainfall-runoff application. The results show that all three NFM models provide better results when compared against HBV with the NFMs achieving CE ranging from 0.88 to 0.90, RMSE 1.19 to 1.38 m³/s, CSI from 0.56 to 0.57, POD from 0.67 to 0.69 and FAR 0.21 to 0.24. This is while the HBV is achieving 0.80 CE, 1.8 m³/s RMSE, 0.43 CSI, 0.65 POD, and 0.44 FAR. Therefore, in terms of the NFMs being able to predict overall fits and meet threshold requirements, these results are acceptable. GSETSK, DENFIS and ANFIS produce almost the same results in terms of goodness of fit indices, however GSETSK achieving lower RMSE and MAE values. Results from the threshold-based indices, adopting a threshold of 15 m³/s (Talei et al., 2013) are illustrated in Figure 4.1(b). The results show better performance of NFMs over HBV model. Figure 4.2 illustrates the results of GSETSK with HBV adopted as a benchmark. The model is able to predict both the high and low flow events well, notwithstanding a value close to false prediction on 15 Jan 1981.

Figure 4.3 shows the results (computed over the entire test period, Table 4.1) for 1, 3 and 5 days ahead water level forecasting at Pakse. In general, the NFMs are able to provide similar or better results compared to the URBS model, which requires more data and time to set up. The results of 1 day ahead forecasting shows almost the same performance for all models in terms of general goodness of fit indices. However in terms of threshold-based indices (11m as recommended by Mekong River Commission (MRC, 2015)) DENFIS has the best performance (CSI= 0.93, FAR= 0.03, POD= 0.95) and URBS has the worst one (CSI= 0.84, FAR= 0.12, POD= 0.95). Also GSETSK (CSI= 0.90, FAR= 0.03, POD= 0.93) is performing better than ANFIS (CSI= 0.86, FAR= 0.05, POD= 0.90). For 3 day ahead forecasting again all of the models have almost the same performance in terms of general goodness of fit indices. However in terms of threshold-based indices, DENFIS (CSI= 0.77, FAR= 0.11, POD= 0.85) have the highest

performance followed with GSETSK (CSI= 0.71, FAR= 0.14, POD= 0.80) and ANFIS (CSI= 0.71, FAR= 0.03, POD= 0.73). Again for this forecasting, URBS has the worst performance (CSI= 0.55, FAR= 0.28, POD= 0.70). For 5 day ahead forecasting, almost all the models have the same performance in terms of general goodness of fit indices. However DENFIS (CSI= 0.72, FAR= 0.17, POD= 0.85) and then GSETSK (CSI= 0.69, FAR= 0.20, POD= 0.78) have the best performances in terms of threshold-based indices. ANFIS (CSI= 0.53, FAR= 0.22, POD= 0.63) is at the next lower of performance followed by URBS (CSI= 0.41, FAR= 0.44, POD= 0.60) which has the worst performance (Figure 4.3(b)).

The above comparisons show that neuro fuzzy models and local models in particular are able to provide results that are at least comparable to the benchmarks. Most notably, both local models require less data and time to train compared to the ANFIS and physically based models. In addition, the local learning feature of DENFIS and GSETSK makes these two models most suitable as real time models as they can be adapted continuously (online) without the need for re-calibration (as in the case of physically-based models) or re-trained (global NFM).

Figure 4.1: Evaluation indices for different models for rainfall-runoff application.

(a) Goodness of fit indices, and (b) Threshold-based indices.

Figure 4.2: Results of 1 day ahead rainfall-runoff forecasts

Figure 4.3: Evaluation indices for river routing application. (a) 1 day ahead, (b) 3 days ahead, (c) 5 days ahead

4.3.3 Rule Base Dynamics

A comparison of the runoff (Figure 4.4(a)) and water level (Figure 4.5(a)) time series for the rainfall-runoff and river routing applications, contrasts the complexities involved between these two processes. The runoff time series is highly complex owing to the complex process of transforming precipitation into runoff from a catchment. In addition, seasonal effects are also present since the entire year's data is used. This is in comparison to the river routing problem where the translation of the flood wave from upstream to downstream is smoothly varying. In addition, since water levels recorded during the flood season are used, seasonal effects are less significant with the Mekong data. Rule base dynamics of the NFMs are analyzed in the context of these processes.

The rule base dynamics for the rainfall-runoff problem is illustrated in Figure 4.4(b). The rule base for ANFIS is static, with the number of rules fixed at 9 rules, established during training using the SC clustering algorithm (Table 4.1). Both the number of rules and their parameters do not change during the test phase. Even for such a complex process as in rainfall-runoff modelling, ANFIS assumes that the same rules can be applied after training. For the DENFIS model, one year of data was used for model training (11 Jan 1961 to 10 Jan 1962) which created an initial rule base of 8 rules. During the test phase (11 Jan 1962 to 31 Dec 1990), the number and parameters associated with the rules increased in response to the complexities of the rainfall-runoff behavior, increasing to 20 rules at the end of 1990. It is further observed that most of the new rules were generated during the first 8 years (roughly 1962 to 1970) as the model encountered new data and was further adapted online. After this period however, the number of rules did not increase further until Dec 1980, Oct 1985 and Sep 1986 where one more rule was added each of these times. After this time the model had a total of 20 rules with no more rules being added until the end of the test period. The rule base in DENFIS thus adapts online, generating new rules which describe ECM's reaction to the changes, whilst during periods where no new rules were added, ECM modifies the existing rules' parameters.

The GSETSK model has the most sophisticated rule base system. As shown in Figure 4.4(b), the dynamic rule base arising from the rule pruning and rule updating procedures allows the model to either increase or decrease the number of rules online, as it responds to the complexities of the rainfall-runoff problem. Similar to DENFIS, GSETSK rule base has an increasing trend in the first decade. Starting from an empty rule base on 11 Jan 1961, the number of rules increased to 11 rules around Sep 1970. As to be expected, during this time, rule construction is observed to be more dominant than rule pruning, after which both rule generation and pruning progress interchangeably, keeping the rule base to a size of approximately eight rules. Interestingly, GSETSK achieves a similar number of rules as ANFIS at the end of the test period; however ANFIS required a significant portion of the data to be used for batch training. It is also observed that the number of rules increased markedly during the extreme peak event in 1980, where the number of rules increases from 7 to 12 over a 7 month period. After this time, lower levels of runoff were encountered and there is less fluctuation in the rule base. This shows the dynamic nature of GSETSK, progressing from an initial stage of rule development, becoming highly active during extreme periods of runoff and stabilizing during less extreme runoff periods. This is in contrast to the DENFIS model, where the number of rules could only increase. Without the removal of rules that may have become redundant or obsolete, the model becomes over parameterized and deterioration in the results can be expected. This observation suggests that a more sophisticated online model such as GSETSK is necessary in order to respond to the dynamic nature of the rainfall-runoff process.

The FAR (Figure 4.4(b)) computed on a yearly basis shows that FAR values for GSETSK are generally higher for the first half of the data series and generally reduces in the latter half. Large spikes in FAR for DENFIS however are evident, even in the second half of the data series (e.g. 1983, 1988, 1990) in spite of the online feature in DENFIS. As mentioned, the number of DENFIS rules rises to approximately 20 at the latter stages, about double the number of GSETSK rules. Presumably, this increase in the number of DENFIS rules makes the rule base unnecessarily complicated resulting in deterioration in results.

Figure 4.4: (a.) Time series of discharge and (b.) Change in the number of rules and yearly FAR during the model operation for different NFMs in the rainfall-runoff forecast application.

As evident in Figure 4.5(a), the water level time series for the river routing model is less complex and more smoothly varying, compared to the rainfall-runoff problem. The rule base dynamics will presumably be different in response to the relative complexity between these two processes. An ANFIS model with nine rules was established using the training dataset after which the ANFIS model was used to provide the water level forecasts of the test dataset. Similar to the rainfall-runoff problem, both the number (9 rules) and parameters were determined during training and did not change during the test phase. The DENFIS model produced six rules at the end of the training period (1 Jun 1994 to 31 Oct 1994). This number of rules was found to be increased to 7 on Sep 1995 and 8 and 9 on Jul 1999. At other times, the number of rules remained fixed throughout the remainder of the model run although model parameters were updated. For GSETSK, the model started with the generation of the first rule on 1 Jun 1994 after which the number of rules fluctuated between 4 to 10. Significantly, the number of rules generated by the three NFMs are similar at the end of the test period for the river routing problem. The FAR (Figure 4.5(b)) values calculated on a yearly basis (FAR values for 1998, 1999, 2009, 2010 are not available since the water levels are less than 11m) shows that generally FAR values for GSETSK and DNEFIS are very similar; however both the local models have significantly smaller FAR values for the 2011 flood season compared to URBS. These observations might imply that the river routing problem is less complex and therefore does not require the need for an NFM with online capabilities.

The above analysis contrasted rule base dynamics of different NFMs for different types of forecast applications. For both the processes studied, it is evident that static models are limited since its rule base remains fixed during the test phase. This can also imply that a global model such as ANFIS will in general require a relatively large training dataset, to increase the likelihood that the training dataset will generate enough global rules. Among the two dynamic models studied, GSETSK accords an obvious advantage for the more complex rainfall-runoff problem, since the model has a more compact and up-to-date rule base. For the river routing problem however, which is simpler, the benefits of GSETSK are modest. In

this case, both DENFIS and GSETSK have roughly the same number of rules at the end of the model operation.

Figure 4.5: (a.) Time series of water level and (b.) Change in the number of rules and yearly FAR during model operation for different NFMs in the river routing forecast application.

4.4 CONCLUSIONS

The following can be concluded from this study:

- (i) All NFMs compared favorably against physical models adopted as benchmark.
- (ii) The local NFMs (DENFIS and GSETSK) require considerably less data for training when compared against global model in order to achieve the same results. In fact while GSETSK and DENFIS use only one sample each time in the online learning, ANFIS uses much more data for its batch training. This is 6929 and 918 data samples in this study for rainfall-runoff and river routing problems respectively.
- (iii) Rainfall-runoff modelling is a complex process and the GSETSK model was found to be best suited for this application as the model was able to generate a compact and up-to-date rule base. The DENFIS model resulted in the generation of a rule base that was much larger in size and presumably more complicated. Although ANFIS resulted in a rule base comparable in size to GSETSK, this came at a cost of having the need for a large dataset for model training.
- (iv) All NFMs produced similar number of rules for the river routing application. River routing is a relatively simpler process to model and our study shows that both the local models were able to achieve similar results, i.e. the process is relatively simpler to model and does not require sophisticated rule base features. Although ANFIS provided similar results to the local models, this came at a cost of having the need for a large dataset for model training.

CHAPTER 5: FLOW FORECASTING IN BASINS WITH LIMITED DATA

5.1 Introduction

DDMs often need to be trained with a dataset which is representative of the system under consideration (Solomatine et al., 2008). Among the variety of DDMs reported in the literature, NFMs have recently gained popularity due to their reasoning and learning capabilities. Studies utilizing NFMs for flow forecasting usually employ either the ANFIS or DENFIS models. These studies were presented in the literature review in Sections 2.2 and 2.3. With NFMs such as ANFIS, each rule is a mapping from a partition of the input space created using a specific clustering method, to a part of the output space. So in the cases where training data is insufficient, the rules may be inadequate to completely describe the input-output space. Also if the testing data deviate significantly from the training data, the NFM rules may not be valid. To resolve these problems, a dynamic model can be used, where instead of having an NFM that is trained using offline learning or batch data, online learning is used. Data is fed to the NFM sequentially one at a time in online learning. Also in online learning there is no priori-knowledge about the size of data which would be available to train the model. Currently, online models such as DENFIS suffer from one or both of the following problems: Firstly, the upper/lower bound of the data used for training is required for model normalization. Secondly, the number of rules increases monotonically. In the latter case, sometimes there is a need to repartition the input space and make new rules. This however cannot be done when the input-output space mapping is saturated with old rules, which may contain rules that are no longer relevant. In addition, the existence of old rules may reduce the accuracy and interpretability of the model. In this chapter the capabilities of GSETSK is evaluated for forecasting in basins with limited data.

5.2 GSETSK Parameter Values Adopted

Detailed information about GSETSK was presented in Section 3.5 and initialization of GSETSK model structural parameters was described in Section 3.5.2 and summarized in Table 5.1. These parameters remain fixed through the simulation period. As shown in Table 5.1, η , thresA , thresB , $\text{WP}(0)$ and thresP were chosen based on values in the literature. The recommended value for θ and λ are 0.4 and 0.99, respectively (Nguyen et al., 2015). However in this study $\theta = 0.6$ and $\lambda = 0.97$ were chosen as this combination of parameters was able to provide greater rule generation diversity. In order to make the model general (at least for the two catchments), however, we have ensured that the same values of θ and λ were adopted for both cases. ν had to be chosen differently between the two cases since ν determines the width of the Gaussian Membership Function, which is dependent on the specific data. The initialization of the consequent parameters (matrix B) is data specific. Depending on the data used, the first data presented was used for initial tuning of the consequent parameters (Eq. 3.31) the model. However, the consequent parameters are continuously updated with the presentation of each subsequent data point.

Table 5.1: GSETSK parameter setting for rainfall-runoff and river routing forecast applications

Study Area	ν	θ	λ	thresA	thresB	WP(0)	η	thresP
Rainfall-Runoff	13	0.6	0.97	0.8	0.7	0.5	0.5	0.5
River Routing	3	"	"	"	"	"	"	"
Reference	-	-	-	Turksen and Zhong, (1988), Nguyen et al., (2015)		(Nguyen et al., 2015)		

5.3 Data Used and Evaluation of Results

5.3.1 Study Area and Data

Data from two different geographical locations were used in this study. The first dataset belongs to Klippan_2 Catchment which was described in Section 3.6.1. Daily data of river discharge (Q), precipitation (P) and temperature (T) from 1 Jan 1979 to 31 Dec 1980 were used for the rainfall-runoff forecasting in this catchment. Temperature was taken as a surrogate parameter for snowmelt. As this is a small catchment (242.9 km²) and hence the time of concentration is low, the main variables lagged at 1 time step were chosen for forecasting 1 time step ahead. In this manner, the next day's discharge (Q₁) is forecasted (see Eq. (5.1)) as a function of current day discharge (Q₀), current day precipitation (P₀), and current day temperature (T₀):

$$Q_1 = f(Q_0, P_0, T_0) \quad (5.1)$$

The second dataset used in this study is from the Lower Mekong River (described in Section 3.6.2). Daily stage data for the monsoon season (Nguyen and Chua, 2012) from June to Oct for two consecutive years (2010 and 2011) containing 271 data points were analyzed. The river stage at two stations namely Pakse (PK) and Savannakhet (SV) were used for a river routing application. Forecasting is done for 1, 3 and 5 days ahead. The flow from Pakse to Savannakhet, located about 240 km downstream, is estimated to take about 1-2 days (Nguyen and Chua, 2012). 1 day ahead forecasting for Pakse can be represented by:

$$PK_1 = f(PK_0, SV_{-1}) \quad (5.2)$$

where PK₁ and PK₀ are next day and current day stage at Pakse Station, respectively, and SV₋₁ is the stage at Savannakhet one day before. 3 day ahead forecasting can be represented by:

$$PK_3 = f(PK_{-2}, SV_{-3}) \quad (5.3)$$

where PK_3 and PK_{-2} are 3 day ahead and 2 days before stage at Pakse Station and SV_{-3} is 3 days before stage at Savannakhet. Similarly, 5 day ahead forecasting can be represented by:

$$PK_5 = f(PK_{-4}, SV_{-5}) \quad (5.4)$$

where PK_5 and PK_{-4} are 5 day ahead and 4 days before stage at Pakse Station and SV_{-5} is 5 days before stage at Savannakhet.

5.3.2 Model Comparisons

GSETSK was used for rainfall-runoff (Swedish Catchment) and river routing (Lower Mekong) forecast applications. The GSETSK model was run fully online using the entire dataset. In this way, the GSETSK model was continuously trained from the 1st data sample and the model was used for forecasting from the 2nd sample onwards. The GSETSK model was run from 1 Jan 79 to 31 Dec 1980 and the monsoon seasons (June to October from 11 June 2010 to 31 Oct 2010 and 14 June 2011 to 20 Oct 2011) for the Swedish catchment and Lower Mekong, respectively. As none of the other NFMs can be applied in this way, only the naïve model and results of the calibrated physically-based models, URBS and HBV, were used for comparison. The naïve model is a statistical model which equates the forecast value to the current observed value. It is therefore a ‘perfect’ forecast, i.e. the forecast value = observed data, albeit lagged by the forecast lead time. It can be expected that GSETSK model used in such a way should produce results better than the naïve model but not better than the calibrated model. Since the assumption here is that the study areas have no prior data, the comparison with the calibrated models may seem superfluous. However, this comparison can be used to provide an indication of the overall accuracy of the GSETSK model in its ability to handle situations where prior data is non-existent. This is the subject of Scenario 1 (forecasting in basins without prior data) presented in Section 4.1. This scenario simulates the situation where a previously ungauged basin is newly equipped with monitoring equipment and the logged data are fed to GSETSK directly after installation.

As the length of the dataset is increased, models which require prior data can be used. This is the subject of Scenario 2 (forecasting in basins with limited data). In this scenario half of the data set is used for training of DENFIS, RBF and MLP, then the forecasts (testing) is carried out using the 2nd half of the dataset. These forecast results were compared against GSETSK model results taken from Scenario 1 which coincides with the testing period of the other models. This comparison thus allows an evaluation of traditional data driven models, in a limited data scenario, against GSETSK which is run fully online. The data used in the training and prediction stages of the DENFIS, RBF and MLP models are summarized in Table 5.2.

Table 5.2: Data used for the training/testing of data-driven models (excluding GSETSK) in Scenario (2): Basins with limited data

Study Area	Function	Start date	End date	No. of data samples
Klippan_2 (rainfall-runoff forecast)	Train	1 Jan 1979	31 Dec 1979	365
	Test	1 Jan 1980	31 Dec 1980	366
Mekong River (river routing forecast)	Train	11 June 2010	31 Oct 2010	143
	Test	14 June 2011	20 Oct 2011	128

5.4 Results and Discussion

5.4.1 Forecasting in Basins without Prior Data (Scenario 1)

Table 5.3(a) shows the forecasting results and evaluation indices for rainfall-runoff forecast with GSETSK for 1979 and 1980 (except the first day which was used for initialization) compared with the naïve model and Table 5.3(b) shows the GSETSK results for 1980 compared with the naïve and HBV models for 1980 (since HBV model results are only available for 1980 (Talei et al. 2013)). The time series are compared in Figure 5.1. Note that the HBV model was calibrated using the daily data from 1 Jan 1961 to 31 Dec 1979 and 1 Jan 1986 to 31 Dec 2003 and requires 1

day ahead forecast of rainfall in order to provide flow forecasts (Talei et al., 2013). These results show that although GSETSK model was able to achieve only moderate improvements of CE and R^2 by 5 % and 4% over the HBV model, this is nonetheless significant as the result is based on a fully online model without the need for prior calibration or data.

Table 5.3: (a) Comparison of GSETSK vs. Naïve model for rainfall-runoff forecast application (1979-1980). (b) Comparison of GSETSK vs. HBV model for rainfall-runoff forecast (1980)

(a)

Model	CE	R^2	RMSE (m ³ /s)	MAE (m ³ /s)
GSETSK	0.89	0.90	1.50	0.61
Naïve	0.86	0.86	1.72	0.70

(b)

Model	CE	R^2	RMSE (m ³ /s)	MAE (m ³ /s)
GSETSK	0.87	0.87	1.89	0.78
HBV	0.82	0.83	2.21	1.36
Naïve	0.82	0.83	2.23	0.98

Figure 5.2 shows the evaluation indices for the Lower Mekong River. The URBS utilizes daily precipitation forecast from Satellite Rainfall Estimate (SRE) which was provided by National Oceanic and Atmospheric Administration (NOAA). The comparison between GSETSK and the URBS model shows that GSETSK and URBS have almost the same performance indices for 1 day forecasts. The error indices indicate that GSETSK provides comparable results to the URBS model. Figure 5.3 compares the time series of the model forecasts. Comparing GSETSK and URBS results in these figures it can be seen that GSETSK generally has smoother forecasted time series. It shows that the online model GSETSK can provide results comparable to URBS.

Figure 5.1: Comparison of GSETSK and HBV model results for rainfall-runoff forecast application for 1980

Figure 5.2: Comparison of GSETSK, URBS and Naïve models for river routing forecast application

(a)

(b)

Figure 5.3: Result of river routing forecast application for Lower Mekong River catchment by (a) GSETSK and (b) URBS

5.4.2 Forecasting in Basins with Limited Data (Scenario 2)

In this scenario, the two basins were treated as basins with limited data. This analysis was carried to benchmark GSETSK against DENFIS, RBF and MLP which require historical data for training. GSETSK model results was taken from Scenario 1 (which coincides with the testing period of the data driven models in Scenario 2) were used for comparison. Table 5.4 compares the results of GSETSK against traditional NFMs for the rainfall-runoff. The results show that GSETSK is able to achieve results that are comparable with other data driven models, even though these models were trained with limited historical data and GSETSK model was operating in a fully online manner. In addition, a model such as the RBF neural network model is an opaque model. Also considering DENFIS, it should be noted that the number of rules in DENFIS increased to eight, since DENFIS does not have a rule pruning algorithm. GSETSK however has a maximum of 4 rules which makes the model more relevant to the current data distribution as well as simpler to comprehend. This aspect is detailed in Section 5.4.3.

Table 5.4: Performance of GSETSK against traditional models for rainfall-runoff application.

Model	CE	R ²	RMSE (m ³ /s)	MAE (m ³ /s)
GSETSK	0.87	0.87	1.89	0.78
RBF	0.87	0.87	1.90	0.81
DENFIS	0.86	0.87	1.98	0.80
MLP	0.82	0.87	2.21	0.83
HBV	0.82	0.83	2.21	1.36
Naïve	0.82	0.83	2.23	0.98

Table 5.5 shows the GSETSK results for the river routing application benchmarked against other data driven models. Similarly, GSETSK is able to achieve comparable results for all 1, 3 and 5 days ahead forecasts. Although GSETSK and DENFIS results are close, it should be mentioned that the DENFIS model requires prior data for training. For the 5 days ahead forecast, DENFIS is

able to achieve a slightly higher CE and R^2 than GSETSK; 0.1 m lower RMSE and 0.04 m lower MAE. However, as will be further elaborated in Section 5.4.3, it should be noted that in terms of interpretability GSETSK is superior to DENFIS, where DENFIS has highly overlapping knowledge base (which is difficult to interpret and comprehend) while GSETSK has a knowledge base that is more distinguishable.

Table 5.5: Performance of GSETSK against traditional models for river forecast application

Model	Forecast lead time (days)	CE	R^2	RMSE (m)	MAE(m)
GSETSK	1	0.98	0.98	0.38	0.26
	3	0.88	0.89	0.85	0.66
	5	0.78	0.79	1.17	0.91
DENFIS	1	0.98	0.98	0.32	0.25
	3	0.90	0.91	0.80	0.63
	5	0.82	0.83	1.07	0.87
URBS	1	0.98	0.98	0.33	0.25
	3	0.89	0.90	0.82	0.65
	5	0.77	0.79	1.19	0.95
RBF	1	0.97	0.98	0.40	0.32
	3	0.87	0.89	0.89	0.73
	5	0.73	0.79	1.29	1.00
MLP	1	0.92	0.97	0.70	0.54
	3	0.75	0.86	1.24	1.00
	5	0.45	0.66	1.85	1.53
Naïve	1	0.98	0.98	0.35	0.26
	3	0.88	0.89	0.87	0.67
	5	0.75	0.78	1.28	1.01

5.4.3 GSETSK Rule Base and Interpretability

For brevity, only the rule base dynamics for the rainfall-runoff application is presented in detail. At the initiation of the model, GSETSK begins with an empty rule base and the first rule is created. GSETSK begins with a simple linear model in the first instance and more rules are created as the model is presented with more data. With reference to Figure 5.6, the 1st rule is established with the presentation of the first data point (1 Jan 1979):

$$Q_1 = FS_1 * (0.008Q_0 + 0.01 P_0 + 0.003 T_0 - 0.097) \quad (5.5)$$

where FS_1 is the firing strength of the 1st rule ($FS_1 = 1$) and Q_0 , P_0 and T_0 are river discharge, precipitation and temperature variables respectively recorded on 1 Jan 1979. The consequent parameters are determined using RLS as detailed in Section 3.5.3. From 1 Jan 1979 to 12 Apr 1979, the number of rules remains at unity but is constantly updated with the presentation of each data point. The model output on 12 Apr 1979 where $FS_1 = 0.66$ is determined by:

$$Q_1 = FS_1 * (0.22Q_0 + 0.92 P_0 + 0.2 T_0 + 0.08) \quad (5.6)$$

A second rule (Rule 2) is added on 13 Apr 1979 and subsequently the model uses two rules for prediction. With $FS_1 = 0.57$ and $FS_2 = 0.99$ determined for Rule 1 and Rule 2, respectively, Q_1 is now:

$$Q_1 = (0.57*(0.21Q_0 + 0.92 P_0 + 0.2 T_0 + 0.07) + 0.98*(0.12Q_0 + 0.63P_0 + 0.07T_0 + 0.3)) / 2 \quad (5.7)$$

The above example demonstrates the incremental procedure where rules are created/modified and though not shown here, deleted (this is a feature of GSETSK only) in response to the presentation of each data point. In this way, a fully-online model such as GSETSK can be used in basins without prior data.

The rule base of GSETSK and DENFIS at the end of the simulation period for the rainfall-runoff forecast application is shown in Figure 5.4. GSETSK has two GMFs for each input dimension. The GMFs for the discharge dimension can be labeled as Low and High while the GMFs for the precipitation dimension can be labeled as

Light and Heavy, and the GMFs in the temperature dimension can be labeled as Cold and Moderate. In contrast, DENFIS has 8 membership functions for every input dimension which makes the model significantly more complex. Please note that the spike in Figure 5.4. means a cluster which is in early stages and has not been expanded. Figure 5.5 shows the knowledge-base for GSETSK and DENFIS for the 1 day ahead forecast model in the river routing forecast application. It clearly shows that GSETSK has distinguishable GMFs which can be readily labeled with clear semantics. In this case, three stages labeled as Low, Medium and High can be assigned to different GMFs in both input spaces, i.e., the current river stage at Pakse Station and the previous day river stage at Savannakhet Station. In comparison, there are twice as many more labels generated in DENFIS than in GSETSK. Having a large number of labels results in highly overlapping and complex structure. On the other hand, the MSGC clustering procedure produces a compact and distinguishable knowledge-base and the rule pruning algorithm helps GSETSK to maintain a more compact and up-to-date rule-base.

The linguistic labels in GSETSK can be used to describe the rules in a more intuitive way and these rules can be linked to the physical system. This can be illustrated from an inspection of the rule-dynamics of the two datasets used. Figure 5.6 shows the rule dynamics of GSETSK during the simulation phase for the rainfall-runoff application.

Figure 5.4: Knowledge-base of different input dimensions at the end of forecasting with DENFIS and GSETSK fuzzy neural modeling (rainfall-runoff forecast)

Figure 5.5: Knowledge-base of different input dimensions at the end of forecasting with DENFIS and GSETSK fuzzy neural modeling (river routing forecast)

Figure 5.6: Rule dynamics of GSETSK for 1 day ahead rainfall-runoff forecast application

There are three main phases detectable. In the first phase, the number of rules increases from one in 1 Jan 1979 to four almost at the end of year 1979. These rules (R1A-R4A) are as follows:

1. R1A: IF Q_0 is Low AND P_0 is Light AND T_0 is Cold THEN $Q_1 = 0.93Q_0 + 0.16P_0 + 0.026T_0 - 0.23$
2. R2A: IF Q_0 is Low AND P_0 is Light AND T_0 is Moderate THEN $Q_1 = 0.92Q_0 + 0.12P_0 - 0.004T_0 - 0.05$
3. R3A: IF Q_0 is Low AND P_0 is Light AND T_0 is Warm THEN $Q_1 = 0.94Q_0 + 0.08P_0 - 0.006T_0 + 0.05$
4. R4A: IF Q_0 is Low AND P_0 is Moderate AND T_0 is Moderate THEN $Q_1 = 0.93Q_0 + 0.15P_0 + 0.019T_0 - 0.19$

A careful inspection of the antecedents of these rules shows that for rules R1A to R3A, the discharge is Low and precipitation is Light. Therefore, temperature is

the main variable which varies from Cold to Moderate and finally to Warm in the rules R1A to R3A respectively. In contrast to other rules, Rule R4A uses a new GMF, i.e. Moderate precipitation. The second phase is approximately from 23 Dec 1979 to 27 Mar 1980. Rules R3A and R4A were deemed as irrelevant and were subsequently deleted by the rule pruning algorithm. These are the rules involving Moderate precipitation (R4A) or Warm temperature (R3A) linguistic labels which are no longer important for the first half of 1980. In the third phase, approximately from 10 June 1980 to 31 Dec 1980, two more rules (R3B and R4B) were created. These rules were created to account for the introduction of the linguistic variables, Heavy for precipitation (R3B) and High for discharge (R4B). Interestingly, the change in rules observed for the different phases can be attributed to changes in season and weather conditions. For rule R1B and R2B however, the main change is in the parameters of the linear function in the consequent part. The following shows the rules at the end of the GSETSK simulation period:

1. R1B: IF Q_0 is Low AND P_0 is Light AND T_0 is Cold THEN $Q_1 = 0.77Q_0 + 0.37P_0 + 0.014T_0 + 0.25$
2. R2B: IF Q_0 is Low AND P_0 is Light AND T_0 is Moderate THEN $Q_1 = 0.85Q_0 + 0.31P_0 - 0.028T_0 + 0.18$
3. R3B: IF Q_0 is Low AND P_0 is Heavy AND T_0 is Moderate THEN $Q_1 = 0.94Q_0 + 0.45P_0 - 0.175T_0 - 0.62$
4. R4B: IF Q_0 is High AND P_0 is Light AND T_0 is Moderate THEN $Q_1 = 0.82Q_0 + 0.34P_0 - 0.023T_0 + 0.15$

Figure 5.7 shows the rule dynamics for the 1 day ahead river routing forecast application. The number of rules increases from one to a maximum of six rules at the end of GSETSK simulation run. Rule pruning does not occur for this database and the knowledge or rule-base shows an increasing trend. This can be due to the fact that the only inputs for this problem are stages of the two stations at different times which are changing smoothly. Furthermore, there are no seasonality effects as all the data were derived from the monsoon season. As the overall trend of stage

time series is ascending, the linguistic variables for the two stations generally change from Low to Medium and High from rules R1 to R6. The following shows the rules for 1 day ahead river stage forecasting in the Lower Mekong River:

1. R1: IF PK_0 is Low AND SV_{-1} is Low, THEN $PK_1 = 0.83PK_0 + 0.17SV_{-1} + 0.17$
2. R2: IF PK_0 is Low AND SV_{-1} is Medium, THEN $PK_1 = 0.79PK_0 + 0.19SV_{-1} + 0.34$
3. R3: IF PK_0 is Medium AND SV_{-1} is Medium, THEN $PK_1 = 0.88PK_0 + 0.13SV_{-1} + 0.02$
4. R4: IF PK_0 is Medium AND SV_{-1} is Low, THEN $PK_1 = 0.9PK_0 + 0.1SV_{-1} + 0.15$
5. R5: IF PK_0 is Medium AND SV_{-1} is High, THEN $PK_1 = 0.82PK_0 + 0.19SV_{-1} + 0.14$
6. R6: IF PK_0 is High AND SV_{-1} is High, THEN $PK_1 = 0.87PK_0 + 0.13SV_{-1} + 0.14$

Figure 5.7: Rule dynamics of GSETSK for 1 day ahead river routing forecast application

5.5 Conclusion

The following can be concluded from this study:

- (v) When GSETSK model results were compared against the physically based models, URBS and HBV, similar results were obtained. However, the obvious advantage of GSETSK is that a separate training phase is not required, since it is a fully online model.
- (vi) Being a fully online model, GSETSK was also found to be advantageous as it does not require data for model initialization. This is in comparison with NFMs such as DENFIS which require prior data since upper/lower bound limits need to be set for data normalization and model initialization.
- (vii) GSETSK is able to provide up-to-date rules dynamically in online applications and the rules can be better related to the physical process of the system under consideration. Thus although similar results were obtained compared to DENFIS, this was achieved with a smaller number of rules. This made the rule base more compact and easier to interpret.

CHAPTER 6: SPARSE FUZZY RULE-BASE FOR FLOW FORECASTING

6.1 Introduction

Limitations in the range and non-uniformity in the distribution of hydrologic data are two problems that are commonly encountered in the use of data driven models. Ideally, the data should be normally distributed, or transformed to normality using an appropriate transformation. Applications of data normalization techniques have been attempted for example by Sudheer et al., (2003) with ANN models. Nayak et al., (2004) have compared the application of ANFIS using transformed versus untransformed data for river flow modeling. The authors compared different transformations and found that the Wilson-Hilferty transformation (Salas et al., 1985) improved training and validation by up to 28% and 11%, respectively. Furthermore, improvements were obtained in peak flow estimation when using normalized data. Other studies applying data transformation with ANFIS for flood forecasting include (Nayak et al., 2007; Nayak et al., 2005b; Rath et al., 2013). With regards to the issue of extrapolation, Fortin et al. (1997) mentioned that data driven models do not have the ability to extrapolate, or make predictions beyond the range of the training data.

Classical NFM's such as ANFIS and DENFIS work well if a dense dataset is available. A dense dataset is well suited to the traditional method for training, i.e. partitioning 2/3 of the dataset for training and 1/3 for testing, which eventually results in a dense fuzzy rule base (FRB). In a dense FRB, rules cover the entire universe of discourse and occupy all input spaces. So for each arrival of data, at least one rule can be fired and used for output generation. A dense FRB however is generated only if the model can be trained with large enough dataset; however, this condition is not always satisfied, for example in applications where there are few data available or in dealing with extreme events such as floods. This may locate newly arrived data in the vacant parts of the input space where rules have not been established. In this case, a new rule is created where only its antecedent part (the

location of the new cluster in the input space) is known. However there is no knowledge of the consequent part of the rule as all of the previous rules have been fired to zero level. So the traditional way of estimating the consequent part by the weighted averaging the fired rules (Eq. 3.34) is not applicable. In this situation a model with sparse FRB is created which has not experienced the outcome for the most recently arrived data. In addition, a sparse FRB may arise as a result of using small clusters. In traditional NFMs, it is often the case that a small number of rules will cover large parts of the input space. However this gives rise to the problem of having general rules covering wide areas of the input space which are able only to estimate the general trends of the system under consideration. The generated model may be sufficient for simple systems but for more complex solution surfaces, a more sophisticated approach is necessary.

To resolve the issues associated with the use of sparse datasets, a fuzzy interpolative/extrapolative reasoning (FIER) algorithm is proposed. The FIER strategy is implemented in GSETSK since GSETSK has many advanced features compared to other NFMs. GSETSK has an incremental learning scheme which is suited to work with online data streams, the model therefore uses the most recent data in the time series to update its knowledge-base. The objectives of this study were to develop an interpolation/extrapolation solution which can handle sparse FRB. This involves the development of a fine-representation FRB which is better able to model infrequently occurring extreme events such as floods. A new NFM, namely GSETSK with fuzzy interpolative/extrapolative reasoning or FIER-GSETSK was tested for hydrological forecasting. This study was conducted in the context of rainfall-runoff and river routing applications.

6.2 Fuzzy Interpolation and Extrapolation

Rules are the basic elements in a FIS and appropriate spatial coverage is essential for FIS performance. Initial FISs used human expertise for rule generation which then progressed to automatic rule generation based on data. For example, classical FISs such as the compositional rule of inference or CRI (Zadeh, 1973), Mamdani-

Larsen (Mamdani and Assilian, 1975; Martin Larsen, 1980), and Takagi-Sugeno (Sugeno, 1985b; Takagi and Sugeno, 1985) used a dense (compact) FRB. Hence for all data samples there is at least one rule which can be activated to a certain level for the fuzzy inference procedure.

A coarse FRB comprising a collection of widely covering rules is commonly employed in most NFM models in flood forecasting. As illustrated in Fig. 6.1, there are two fuzzified input spaces X_1 (the horizontal axis) and X_2 (the vertical axis) each including 2 GMFs. The GMFs in each input space are wide and consequently their respective rules cover a wide area creating a coarse FRB. Each rule has a consequent part which can be represented as a plane in n-D space. The model output is decided by the collection of planes activated to a prescribed degree (Eq. 3.17). When the FRB is coarse, the number of planes which describe the solution surface is limited. For complex surfaces, these planes may not be sufficient and more localized planes are required. Thus, having a limited number of widely covering planes may not be entirely suitable for modeling low frequency events such as floods. As an attempt to resolve this problem, a fine FRB (collection of narrow coverage rules) (Figure 6.2) is presented. This model uses narrower GMFs and hence localized output space which generates an opportunistic model which is better able to deal with low-frequency events. However, the following issues arise when using a fine resolution FRB: (i) The FRB becomes sparse and the NFM cannot generate a solution for data that fall in the sparse spaces of the FRB. (ii) In models such as GSETSK which incorporate a pruning algorithm, the sparsity and knowledge loss on low-frequency events are accentuated. This is the result of removing rules that have not been activated for a while e.g. rules associated with low frequency extreme events. In applications such as floods, rules associated with extreme water levels will not be activated frequently, but however, are not obsolete. If removed, this will result in a loss of knowledge by the model which will be critical high flow events. (iii) In the fine-grain-represent FRB, more rules are created which may deteriorate the interpretability of the NFM.

Figure 6.1: Coarse-grain-represent of FRB in GSETSK

Figure 6.2: Fine-grain-represent of FRB in FIER-GSETSK

A sparse rule base may arise for a variety of reasons. (i) Missing or lack of data, (ii) Change to the physical process such that existing rules are no longer valid e.g. the widening of the range of fuzzy input space by an extension of the upper and/or lower bounds of the dataset, and (iii) Omitting redundant rules to deal with the problem of "curse of dimensionality" (Kóczy and Hirota, 1991). The Sparse Fuzzy Model Identification (SFMI) MATLAB Toolbox (Johanyak, 2008) (available at <http://fri.gamf.hu/>) was created for rule generation of less complex models. The motivation for this research is to utilize a fine-grain-represent FRB to have a better representation of the surface function. This approach seems to be useful especially for complex surfaces. To best of our knowledge, this research is the first work on sparcification of FRB for dealing with complex surfaces.

A fuzzy rule interpolation/extrapolation (FRI/E) approach will be adopted for this study. The initial work on this approach was introduced by (Kóczy and Hirota, 1991, 1993a, b). Thereafter, many studies have been conducted in this field. Due the different techniques used for FRI/E, Tikk et al., (2011) suggested the criteria and characteristics (13 properties) for an ideal FRI/E technique. Johanyák et al., (2006) created the Fuzzy Rule Interpolation (FRI) MATLAB Toolbox (available at <http://fri.gamf.hu/>) which includes more than 10 different FRI techniques. Also Yang and Shen (2013) made a classification of different FRI/E techniques. Other interpolation methods can be found in (Johanyák and Kovács, 2006; Perfilieva et al., 2012). In FRI/E, "given any observation and rule base, the interpolated result should remain in general as a fuzzy set" (Yang and Shen, 2013). This is similar to the "shape invariance" property (Tikk et al., 2011) illustrated in Figure 6.3 as an example of a FRI/E technique. It can thus be said that FRI/E techniques are suitable for Mamdani FISs where the output is fuzzy. As the GSETSK model uses a TSK method, a FRI/E technique was designed and implemented in a TSK FIS, which is detailed in the next section. To the best of our knowledge no studies have been conducted to implement FRI/E in a TSK FIS.

(a)

(b)

Figure 6.3: An example of FRI (Closed form FRI) (a) Interpolation procedure, (b) Interpolation results (adapted from Yang and Shen (2013))

6.3 Proposed FIER-GSETSK Model

A new model FIER-GSETSK, incorporating a Fuzzy Interpolating and Extrapolating Reasoning (FIER) algorithm in the GSETSK NFM is proposed. Although FIER can in essence be incorporated into any online NFM, GSETSK was chosen because of its advantages over other NFMs, as GSETSK is a fully-online model which does not require any prior-knowledge on data and it generates an up-to-date compact FRB which is easier to interpret. The incorporation of FIER in GSETSK will enable the model to cope with sparse FRBs. The rule pruning algorithm in GSETSK is deactivated to prevent the loss of important rules and knowledge. Modifications (blue parts in Figure 6.4) to the GSETSK code originally developed by Nguyen (2012) (Section 3.5) in the implementation of FIER-GSETSK include:

- (i) Narrower GMF widths are adopted when a new rule is created. This result in a fine-grain-represent FRB and eventually the sparse FRB.
- (ii) Deactivation of rule pruner to avoid knowledge loss which is more possible in sparse FRB.
- (iii) Identifying the most important rules as the active rules and using them in reasoning and parameter learning phase. This increases the usage of the most relevant cluster as well as improving the interpretability of the model.
- (iv) Using the FIER algorithm for computing the consequent parameters of the newly created rule.

The interpolation/extrapolation should be implemented by using the two rules closest to the most recent data. Determining the two closest rules, their output spaces (which are in a form of two planes in case of 2D input spaces or two hyperplanes in case higher dimension input spaces) are used. Each plane/hyperplane has $n+1$ parameters where n is the input space dimensions (equal to the number of input variables) and the $(n+1)^{\text{th}}$ parameter is the distance from the origin to plane/hyperplane.

Figure 6.4: Modifications (highlighted in blue) to GSETSK original flowchart

Figure 6.5: The linear output space in a TSK NFM and its parameters (2-D input space example)

Figure 6.5 illustrates a surface function related to a 2-D input space where b_0 , b_1 , b_2 are the distances from the origin to the plane, plane dip in x axis and plane dip in y axis respectively. The FIER algorithm searches through the rule-base to identify the two nearest rules to the newly arrived data (red circle in Figure 6.6). These two

rules (the blue circles in Figure 6.6) are then projected on each input space to reveal the two GMFs which should be used. By inspecting the position of the current data (red GMFs) with the GMFs of the two rules (blue GMFs), either interpolation or extrapolation will be implemented. If the data point is located between two existing GMFs, interpolation is performed; else, if the new data point is located beyond the two existing GMFs, extrapolation is performed. Further details of FIER are as follows:

- (i) When the newly arrived data falls in the part of the input space which is not covered by any other rule, FIER is implemented. There should be more than two rules in the rule base in order that FIER can utilize the rule positions for its operation. As illustrated in Figure 6.6 the newly arrived data is located in the part of a 2-D space which is not covered by any other rule. And since there are more than 1 rule (11 rules as indicated by the black circles in Figure 6.6) in the rule base, FIER can be implemented.

- (ii) The Euclidean distance $\|\mathbf{x} - \mathbf{y}\|$

$$\|\mathbf{x} - \mathbf{y}\| = \left(\sum_{i=1}^q |x_i - y_i|^2 \right)^{1/2} / q^{1/2}, \mathbf{x}, \mathbf{y} \in \mathbf{R}^q \quad (6.1)$$

of the newly created rule from all existing rules is computed in the input space, between two q-dimensional rules $\mathbf{x}$ and $\mathbf{y}$, x_i and y_i are the i^{th} component of rules $\mathbf{x}$ and $\mathbf{y}$, respectively. The rules are normalized in all their q dimensions before being used in Eq. (6.1). Normalization ([0.1, 0.9]) is done using the last seen data lower/upper bound ([Min, Max]):

$$V' = 0.8 \frac{V - Min}{Max - Min} + 0.1 \quad (6.2)$$

where V is actual value and V' is normalized value.

- (iii) The two nearest rules to the newly created rule (Figure 6.6) are marked using the Euclidean distance determined from the last step.

- (iv) The GMFs of the two nearest rules and the new created rule are investigated in each input space (Figure 6.6). If both GMFs flank at one side of the GMF of the new created rule, then extrapolations is required. Otherwise if the GMF of the newly created rule falls between the GMFs of the other rules, interpolation is required.
- (v) Linear interpolation or extrapolation is performed by the gradient-based method (Figure 6.7) using the consequent parameter associated with each rule (Eq. (6.3)-(6.6)).

$$b_{13} = D_2 \left(\frac{b_{11} - b_{12}}{D_1 + D_2} \right), \quad b_{14} = b_{12} + b_{13} \quad (\text{Figure 6.7a}) \quad (6.3)$$

$$b_{13} = D_1 \left(\frac{b_{12} - b_{11}}{D_1 + D_2} \right), \quad b_{14} = b_{13} + b_{11} \quad (\text{Figure 6.7b}) \quad (6.4)$$

$$b_{13} = D_1 \left(\frac{b_{11} - b_{12}}{D_2 - D_1} \right), \quad b_{14} = b_{11} + b_{13} \quad (\text{Figure 6.7c}) \quad (6.5)$$

$$b_{13} = D_1 \left(\frac{b_{12} - b_{11}}{D_2 - D_1} \right), \quad b_{14} = b_{11} - b_{13} \quad (\text{Figure 6.7d}) \quad (6.6)$$

where D_1 is the distance of the new created rule from the 1st nearest rule (the center of GMFs of two rules in specific input dimension are used), D_2 is the distance of the new created rule from the 2nd nearest rule (the center of GMFs of two rules in specific input dimension are used), b_{11} is consequent parameter of the 1st nearest rule, b_{12} is the consequent parameter of the 2nd nearest rule, b_{13} is the auxiliary consequent parameter as shown in Figure 6.7 and b_{14} is the consequent parameter of the new created rule

Figure 6.6: The proposed FIER (2-D input space example)

Figure 6.7: Fuzzy rule: (a), (b) interpolation and (c), (d) extrapolation.

6.4 Results

6.4.1 Rainfall-Runoff

FIER-GSETSK was used for rainfall-runoff modeling with the Klippan_2 catchment data where discharge (Q), precipitation (P) and temperature (T) were the candidate input variables for runoff forecasting. Temperature was taken as a surrogate parameter for snowmelt. As the time of concentration is expected not to be in excess of 1 day, the main variables were lagged at 1 time step to forecast flows at 1 time step ahead. In this manner, the next day's discharge (Q_1) is forecasted as a function of current day discharge (Q_0), current day precipitation (P_0), and current day temperature (T_0) (Eq. (5.1)). The data used in this study ranges from 1 Jan 1979 to 31 Dec 1980. The model was run fully online where the 1st input-output data sample was used only for model initialization (there is no testing for that sample) and from the 2nd sample onwards, the model was used for forecasting. With the arrival of each new data, the model was adapted and then used to forecast the next result and so on. This process continued for whole dataset repeating itself with the arrival of each data point. Figure 6.8 shows the results of the modeling with FIER-GSETSK for 1 day ahead runoff forecasts.

Two different types of evaluation indices including general goodness of fit (detailed in Section 3.7.1) and threshold-based indices (detailed in Section 3.7.2) were used for evaluation of the model. Figure 6.9 show the general goodness of fit indices for the whole time series (1979 - 1980). These results are benchmarked against the GSETSK model and it was found that GSETSK is about 3% higher in CE and 0.2 m³/s lower in RMSE. As results of the physically-based HBV model are available for 1980, the evaluation was also performed for this year. Figure 6.10 shows the results of this evaluation. FIER-GSETSK and GSETSK are similar, with both models providing better results compared to the calibrated HBV model. Results using threshold-based indices are shown in Figure 6.11 for 1979 to 1980. The results show improvements of FIER-GSETSK over GSETSK by 3% and 4% for CSI and FAR, respectively. The threshold-based evaluation indices were also computed for the shorter time series (1980) where the results of physically-based model HBV are available (Figure 6.12). FIER-GSETSK results are in general

improved over all other models and it is 9% higher in CSI, 11% lower in FAR compared to HBV.

Figure 6.8: FIER-GSETSK forecasting results for rainfall-runoff modeling

Figure 6.9: General goodness of fit indices for rainfall-runoff modeling (Years 1979-1980)

Figure 6.10: General goodness of fit indices for rainfall-runoff modeling (Year 1980)

Figure 6.11: Comparing threshold based indices of different models for rainfall-runoff modeling (Years 1979-1980)

Figure 6.12: Comparing threshold based indices of different models for rainfall-runoff modeling (Year 1980)

Table 6.1 illustrates the implementation of FIER strategy. In this table the data sample and the date where the FIER model was activated are illustrated. Also for each event, the two closest rules and the specific strategy (interpolation or extrapolation) dealing with the sparse FRB are indicated. There are in total 31 events when FIER was implemented. In each operation one, two or three dimensions (runoff, precipitation and temperature, respectively) were involved. For 10 events, only one dimension was used out of which 7 were interpolations and 3 were extrapolations. For 20 events, FIER was in operation in two dimensions with 7 interpolation events occurring in one of the dimensions and three extrapolations for the other and 11 interpolation events occurring in both dimensions as well as 2 events in which extrapolation occurred in both dimensions. Finally there is only one event with interpolation or extrapolation in all dimensions. Having a closer look at each dimension separately revealed that the most FRI/E occurred for precipitation (dimension 2) and temperature (dimension 3). There were 24 FRI/E operation in precipitation space from which 13 and 11 were interpolation and extrapolation respectively. In temperature space there were 22 FRI/E action which 20 were interpolations and 2 extrapolations. Finally there are 7 FRI/E actions in runoff (dimension 1) from which 5 are interpolations and 2 extrapolations.

Table 6.1: FIER strategy adopted in FIER-GSETSK for rainfall-runoff modeling

Date	2 Nearest Rules	FIER Strategy		
		Dim1 (Q)	Dim2 (P)	Dim3 (T)
9-Jan-79	2 & 1	N/A	Ext	Ext
10-Jan-79	3 & 2	N/A	Int	N/A
1-Apr-79	5 & 4	Int	Ext	N/A
3-Apr-79	6 & 5	N/A	Int	N/A
5-Apr-79	5 & 7	Int	Ext	N/A
13-Apr-79	3 & 8	Int	N/A	N/A
25-Apr-79	9 & 4	N/A	Int	Int
30-Apr-79	2 & 10	N/A	Ext	Int
15-May-79	9 & 10	N/A	Ext	N/A
18-May-79	10 & 12	N/A	Ext	Int
2-Jun-79	12 & 13	N/A	Ext	N/A
1-Jul-79	11 & 2	N/A	Ext	Ext
20-Jul-79	15 & 13	N/A	Int	Int
26-Nov-79	11 & 2	N/A	Int	Int
3-Dec-79	9 & 8	Int	N/A	Int
12-Jun-80	13 & 14	N/A	Int	Int

Date	2 Nearest Rules	FIER Strategy		
		Dim1 (Q)	Dim2 (P)	Dim3 (T)
13-Jun-80	16 & 11	N/A	Int	Int
16-Jun-80	20 & 11	N/A	N/A	Int
18-Jun-80	12 & 18	Int	N/A	Int
19-Jun-80	22 & 18	N/A	N/A	Int
29-Jun-80	11 & 20	N/A	N/A	Int
17-Jul-80	20 & 24	N/A	Int	Int
29-Oct-80	11 & 20	N/A	N/A	Int
30-Oct-80	6 & 5	N/A	Ext	N/A
19-Nov-80	26 & 23	N/A	Int	Int
23-Nov-80	28 & 23	N/A	Int	Int
12-Dec-80	27 & 6	N/A	Ext	Int
13-Dec-80	30 & 5	Ext	Int	Int
15-Dec-80	6 & 27	N/A	Ext	Int
25-Dec-80	26 & 24	Ext	Int	N/A
26-Dec-80	5 & 32	N/A	Int	Int

* Int: Interpolation, Ext: Extrapolation, N/A: Not applicable

6.4.2 River Routing

FIER-GSETSK was also undertaken for river routing modeling in Lower Mekong River. The river stages at two stations namely Pakse (PK) and Savannakhet (SV) were used for this purpose. This contains the daily stage data for the monsoon season (MRC, 2015b) from June to Oct for two consecutive years (2010 and 2011) containing 271 data points. The flow from Savannakhet to Pakse, located about 240 km downstream, is estimated to take about 1-2 days (Nguyen and Chua, 2012). Accordingly, 1 day ahead forecasting for Pakse was conducted using PK_0 (current day stage at Pakse) and SV_{-1} (one day before stage at Savannakhet). For 3 day ahead forecasting, 2 days before stage at Pakse (PK_{-2}) and 3 days before stage at Savannakhet (SV_{-3}) were used. Similarly, for 5 day ahead forecasting, 4 days before stage at Pakse (PK_{-4}) and 5 days before stage at Savannakhet (SV_{-5}) were used. The model was run fully online where the 1st input-output data sample was used only for model initialization (there is no testing for that sample) and from the 2nd sample onwards, the model was used for forecasting. With the arrival of each new data, the model was adapted and then used to forecast the next result and so on. This process continued for whole dataset repeating itself with the arrival of each data point. Figure 6.13 shows the results of the modeling with FIER-GSETSK.

Two different types of evaluation indices including general goodness of fit (Section 3.7.1) and threshold-based indices (Section 3.7.2) were used for model evaluation. Figure 6.14 shows the general goodness of fit indices for whole time series. These results are compared against GSETSK and URBS models. All models provide similar results for 1 day ahead forecasts. For 3 day ahead, FIER-GSETSK and URBS have similar results and for 5 day ahead, FIER-GSETSK and GSETSK have similar results and better than the URBS model.

Threshold-based error indices, which arguably are better indicators of performance for analysis of extreme events as opposed to goodness of fit measures, are shown in Figure 6.15. These indices were calculated for the monsoon seasons for years 2010 to 2011. As illustrated in Figure 6.15, FIER-GSETSK CSI and POD results are better than all of the other models for all 1-3-5 day ahead forecasts. In terms of FAR, FIER-GSETSK results are better than GSETSK for 3 days and 5

days forecasts. Finally Table 6.2 illustrates the FIER strategy taken for the river routing application. In this table the data sample and the date where the FIER was activated are shown. Also for each event, the two closest rules and the specific strategy (interpolation or extrapolation) dealing with the sparse FRB are included. There are 6 FIER implementation for 1 day ahead forecasting and 7 implementations for each of 3 and 5 day ahead forecast. In some cases FIER is implemented as interpolation or extrapolation in only one dimension e.g. on 22 Aug 2010 (for 1 day ahead forecasting) FIER is implemented as an extrapolation in dimension 1. In the other cases, FIER is implemented in both dimensions as two interpolations, two extrapolations or an interpolation and extrapolation in each of the dimensions. For example for 29 Jun 2011 (for 3 days ahead forecast), interpolation occurs in both dimensions. As illustrated in the table, the FIER strategy taken is extrapolation for one or both dimensions for the first implementation for all lead time forecasting. Extrapolation occurs for dimension 2 (Savannakhet Stage).

Figure 6.13: Comparison of FIER-GSETSK with measured data for (a.) 2010, (b.) 2011.

Figure 6.14: General goodness of fit indices for (a) 1 day ahead, (b) 3day ahead, (c) 5 day ahead forecasting in Mekong River

Figure 6.15: 1-3-5 day ahead forecasting results of different models for Mekong River

Table 6.2: FIER strategy taken for different dimensions when using FIER-GSETSK for 1-3-5 day ahead forecasting in Mekong

Time Step	Instant	Date	2 Nearest Rules	FIER Strategy	
				Dim1 (Pakse stage)	Dim2 (Savannakhet Stage)
1 day ahead	73	22-Aug-10	3 & 2	Ext*	N/A*
	93	11-Sep-10	5 & 3	Int*	Int
	129	17-Oct-10	3 & 2	Int	N/A
	143	31-Oct-10	7 & 1	Int	Int
	144	14-Jun-11	8 & 1	Int	Int
	194	4-Aug-11	5 & 6	N/A	Ext
3 day ahead	74	23-Aug-10	2 & 1	Ext	Ext
	76	25-Aug-10	3 & 2	Int	N/A
	132	20-Oct-10	2 & 3	Int	N/A
	158	29-Jun-11	1 & 5	Int	Int
	197	7-Aug-11	4 & 3	N/A	Ext
	200	10-Aug-11	7 & 4	N/A	Ext
	234	13-Sep-11	4 & 8	Int	Ext
5 day ahead	76	25-Aug-10	2 & 1	Ext	Ext
	78	27-Aug-10	3 & 2	Int	N/A
	134	22-Oct-10	2 & 3	Int	N/A
	160	1-Jul-11	5 & 1	Int	Int
	199	9-Aug-11	4 & 3	N/A	Ext
	202	12-Aug-11	7 & 4	N/A	Ext
	236	15-Sep-11	4 & 8	Int	Ext

* Int: Interpolation, Ext: Extrapolation, N/A: Not applicable

6.5 Conclusion

The following can be concluded from this chapter:

- (i) FIER-GSETSK and GSETSK have similar or better results when comparing against physically-based models. The superiority of these NFMs over the physically-based models is more obvious specially when using threshold-based indices.
- (ii) The proposed FIER-GSETSK has better performance than GSETSK in terms of threshold-based indices which may be related to better localization of this model. Although there is a minor deterioration in the generalization ability of FIER-GSETSK reflected in general goodness of fit indices specifically for rainfall-runoff modeling.

CHAPTER 7: CONCLUSION

Forecasting with limited data and sparse data are the main challenges addressed in this study. Using the online neuro-fuzzy model, GSETSK, its capabilities for real-time flow forecasting were investigated in this thesis. Specific features of the model GSETSK in relation to two common hydrological forecast applications, rainfall-runoff and river routing, were elucidated. Studies conducted to investigate fully online application of NFMs in basins with limited data, rule-based dynamics and handling of sparse FRB in a TSK type NFM; The studies which their results have been presented in the specific chapters. A summary of the broad conclusions achieved in this thesis followed by suggestions for future works is presented here.

7.1 Summary

Local NFMs such as DENFIS and GSETSK are suitable for real-time flow forecasting. Results from this study show that similar or improved results, when compared against physically-based models or global NFMs are achievable without the need for model re-calibration/re-training and with lesser data. In terms of interpretability, the local NFM, GSETSK was found to provide a more up-to-date and compact rule base making it suitable for rainfall-runoff forecasting, which is more complex compared to flow routing.

Forecasting with limited data can be handled using GSETSK. The fully-online property of this model, makes it independent from any priori-knowledge requirement e.g. data upper/lower bound for normalization. Also GSETSK online clustering and optimization techniques make it able to adopt any changes in the data. So the model will be capable for forecasting starting from the scratch and handling the changes of system under consideration. GSETSK was found to be a good candidate for forecasting in basins with limited data by providing similar results to current physical models in use in each catchment. The forecasting results are more significant for the complex problem of rainfall-runoff forecasting. Also

GSETSK provide compact an up-to-data rule base which can be easily related to the physics of the process.

Sparse data can be handled by the proposed interpolative/extrapolative reasoning (FIER) algorithm. Using threshold based indices, it was confirmed that more expert rules were created which provided better forecast of infrequent events.

7.2 Future Research Directions

Broadening the usage of GSETSK in water resources engineering applications: The current study has focused on rainfall-runoff modeling and river routing applications. Other applications such as rainfall forecasting, dam reservoir regulation etc. should be studied. This eventually reveals the generalization ability of GSETSK.

Online input selection: Traditional techniques of input selection are offline and assume data availability for input selection. It is proposed that an online input selection method be developed and embedded in an online NFM such as GSETSK. In this case the online input selection model can track the changes in the data and update the model to cope better with time variant phenomena such as seasonality.

Online optimization of the clusters: Current clustering methods are based on unsupervised learning schemes. In models such as ANFIS, clusters are optimized using the gradient descent algorithm; however, an optimization schemes has not been implemented in online NFM GSETSK. It is proposed to develop and embed an online algorithm in the main NFM for optimizing the clusters by their location and spatial coverage.

REFERENCES

- Angelov, P. P., and Filev, D. P., 2004, An Approach to Online Identification of Takagi-Sugeno Fuzzy Models: IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics, v. 34, no. 1, p. 484-498.
- Aqil, M., Kita, I., Yano, A., and Nishiyama, S., 2007, A comparative study of artificial neural networks and neuro-fuzzy in continuous modeling of the daily and hourly behaviour of runoff: Journal of Hydrology, v. 337, no. 1-2, p. 22-34.
- Bardossy, A., 2008, Fuzzy Rule-Based Flood Forecasting, *in* Singh, V. P., Abrahart, R. J., Anderson, M., Bengtsson, L., Cruise, J. F., Kothyari, U. C., See, L. M., Serrano, S. E., Solomatine, D. P., Stephenson, D., Strupczewski, W. G., and SpringerLink (Online service), eds., Practical Hydroinformatics Computational Intelligence and Technological Developments in Water Applications: Berlin, Heidelberg, Springer Berlin Heidelberg.
- Bergstrom, S., 1995, Chapter 13: The HBV model, *in* Singh, V. P., ed., Computer models of watershed hydrology, Water Resources Publications, Littleton, Colo.
- Bergstrom, S., and Forsman, A., 1973, DEVELOPMENT OF A CONCEPTUAL DETERMINISTIC RAINFALL-RUNOFF MODEL: Hydrology Research, v. 4, no. 3, p. 147-170.
- Box, G. E. P., Jenkins, G. M., and Reinsel, G. C., 2008, Time series analysis : forecasting and control, Hoboken, N.J., John Wiley, Wiley series in probability and statistics, xxiv, 746 p. p.:
- Carrol, D. G., 2007, The manual for URBS – A Rainfall Runoff Routing Model for Flood Forecasting & Design (<http://www.urbs.com.au/>).
- Chang, F.-J., and Chen, Y.-C., 2001, A counterpropagation fuzzy-neural network modeling approach to real time streamflow prediction: Journal of Hydrology, v. 245, no. 1-4, p. 153-164.

- Chang, F. J., Chang, L. C., and Wang, Y. S., 2007, Enforced self-organizing map neural networks for river flood forecasting: *Hydrological Processes*, v. 21, no. 6, p. 741-749.
- Chang, L. C., Chang, F. J., and Wang, Y. P., 2009, Auto-configuring radial basis function networks for chaotic time series and flood forecasting: *Hydrological Processes*, v. 23, no. 17, p. 2450-2459.
- Chau, K. W., Wu, C. L., and Li, Y. S., 2005, Comparison of several flood forecasting models in Yangtze River: *Journal of Hydrologic Engineering*, v. 10, no. 6, p. 485-491.
- Chen, C. S., Chen, B. P. T., Chou, F. N. F., and Yang, C. C., 2010, Development and application of a decision group Back-Propagation Neural Network for flood forecasting: *Journal of Hydrology*, v. 385, no. 1-4, p. 173-182.
- Chen, C. S., Jhong, Y. D., Wu, T. Y., and Chen, S. T., 2013a, Typhoon event-based evolutionary fuzzy inference model for flood stage forecasting: *Journal of Hydrology*, v. 490, p. 134-143.
- Chen, P. A., Chang, L. C., and Chang, F. J., 2013b, Reinforced recurrent neural networks for multi-step-ahead flood forecasts: *Journal of Hydrology*, v. 497, p. 71-79.
- Chen, S. H., Lin, Y. H., Chang, L. C., and Chang, F. J., 2006, The strategy of building a flood forecast model by neuro-fuzzy network: *Hydrological Processes*, v. 20, no. 7, p. 1525-1540.
- Chidthong, Y., Tanaka, H., and Supharatid, S., 2009, Developing a hybrid multi-model for peak flood forecasting: *Hydrological Processes*, v. 23, no. 12, p. 1725-1738.
- Chiu, S. L., Cluster estimation method with extension to fuzzy model identification, *in Proceedings IEEE International Conference on Fuzzy Systems* 1994, Volume 2, p. 1240-1245.
- Collischonn, W., Allasia, D., da Silva, B. C., and Tucci, C. E. M., 2007, The MGB-IPH model for large-scale rainfall-runoff modelling: *Hydrological Sciences Journal*, v. 52, no. 5, p. 878-895.

- De Roo, A. P. J., Wesseling, C. G., and Van Deursen, W. P. A., 2000, Physically based river basin modelling within a GIS: The LISFLOOD model: Hydrological Processes, v. 14, no. 11-12, p. 1981-1992.
- Demuth, H., Beale, M., and Hagan, 2009, Neural Network Toolbox™ 6 User's Guide.
- Evans, D., and Jones, A. J., 2002, A proof of the Gamma test: Proceedings of the Royal Society a-Mathematical Physical and Engineering Sciences, v. 458, no. 2027, p. 2759-2799.
- Firat, M., and Güngör, M., 2007, River flow estimation using adaptive neuro fuzzy inference system: Mathematics and Computers in Simulation, v. 75, no. 3-4, p. 87-96.
- , 2008, Hydrological time-series modelling using an adaptive neuro-fuzzy inference system: Hydrological Processes, v. 22, no. 13, p. 2122-2132.
- Firat, M., and Turan, M. E., 2010, Monthly river flow forecasting by an adaptive neuro-fuzzy inference system: Water and Environment Journal, v. 24, no. 2, p. 116-125.
- Fortin, V., Ouarda, T. B. M. J., and Bobée, B., 1997, Comment on "The use of artificial neural networks for the prediction of water quality parameters" by H. R. Maier and G. C. Dandy: Water Resources Research, v. 33, no. 10, p. 2423-2424.
- Govindaraju, R. S., 2000, Artificial neural networks in hydrology. I: Preliminary concepts: Journal of Hydrologic Engineering, v. 5, no. 2, p. 115-123.
- Govindaraju, R. S., and Artific, A. T. C. A., 2000, Artificial Neural Networks in Hydrology. Ii: Hydrologic Applications: Journal of Hydrologic Engineering, v. 5, no. 2, p. 124-137.
- Gupta, A., 2007, Large rivers : geomorphology and management, Chichester, England ; Hoboken, NJ : John Wiley, c2007.
- Hagan, M. T., Demuth, H. B., and Beale, M. H., 1996, Neural network design, Boston, : PWS Pub.
- Han, D., Kwong, T., and Li, S., 2007, Uncertainties in real-time flood forecasting with neural networks: Hydrological Processes, v. 21, no. 2, p. 223-228.

- Hayes, M. H., 1996, Statistical digital signal processing and modeling, New York, John Wiley & Sons, xv, 608 p. p.:
- Haykin, S. S., 1999, Neural Networks: A Comprehensive Foundation, Prentice Hall.
- Hong, Y. S. T., and White, P. A., 2009, Hydrological modeling using a dynamic neuro-fuzzy system with on-line and local learning algorithm: Advances in Water Resources, v. 32, no. 1, p. 110-119.
- Hornik, K., 1991, Approximation Capabilities of Multilayer Feedforward Networks: Neural Networks, v. 4, no. 2, p. 251-257.
- Jang, J. S. R., 1993, Anfis - Adaptive-Network-Based Fuzzy Inference System: Ieee Transactions on Systems Man and Cybernetics, v. 23, no. 3, p. 665-685.
- Johanyak, Z. C., Sparse Fuzzy Model Identification Matlab Toolox - RuleMaker Toolbox, *in* Proceedings Computational Cybernetics, 2008. ICCC 2008. IEEE International Conference on 27-29 Nov. 2008 2008, p. 69-74.
- Johanyák, Z. C., and Kovács, S., 2006, A brief survey and comparison on various interpolation based fuzzy reasoning methods: Acta Polytechnica Hungarica, v. 3, no. 1, p. 91-105.
- Johanyák, Z. C., Tikk, D., Kovács, S., and Wong, K. W., Fuzzy rule interpolation matlab toolbox - FRI toolbox, *in* Proceedings IEEE International Conference on Fuzzy Systems 2006, p. 351-357.
- Juang, C. F., and Lin, C. T., 1998, An on-line self-constructing neural fuzzy inference network and its applications: IEEE Transactions on Fuzzy Systems, v. 6, no. 1, p. 12-32.
- Kang, M. S., Kang, M. G., Park, S. W., Lee, J. J., and Yoo, K. H., 2006, Application of Grey model and artificial neural networks to flood forecasting: Journal of the American Water Resources Association, v. 42, no. 2, p. 473-486.
- Kasabov, N. K., and Song, Q., 2002, DENFIS: Dynamic evolving neural-fuzzy inference system and its application for time-series prediction: IEEE Transactions on Fuzzy Systems, v. 10, no. 2, p. 144-154.
- Kóczy, L. T., and Hirota, K., Rule interpolation in approximate reasoning based fuzzy control, *in* Proceedings Proc. of 4th IFSA World Congress, Brussels, Belgium, 1991, p. 89-92.

- , 1993a, Approximate reasoning by linear rule interpolation and general approximation: *International Journal of Approximate Reasoning*, v. 9, no. 3, p. 197-225.
- , 1993b, Interpolative Reasoning with Insufficient Evidence in Sparse Fuzzy Rule Bases: *Information Sciences*, v. 71, no. 1-2, p. 169-201.
- Kohavi., R., A study of cross-validation and bootstrap for accuracy estimation and model selection., *in Proceedings International Joint Conference on AI 1995*, p. 1137-1145.
- Leahy, P., Kiely, G., and Corcoran, G., 2008, Structural optimisation and input selection of an artificial neural network for river level prediction: *Journal of Hydrology*, v. 355, no. 1-4, p. 192-201.
- Leng, G., Prasad, G., and McGinnity, T. M., 2004, An on-line algorithm for creating self-organizing fuzzy neural networks: *Neural Networks*, v. 17, no. 10, p. 1477-1493.
- Liong, S. Y., Lim, W. H., Kojiri, T., and Hori, T., 2000, Advance flood forecasting for flood stricken Bangladesh with a fuzzy reasoning method: *Hydrological Processes*, v. 14, no. 3, p. 431-448.
- Liu, Z., Martina, M. L. V., and Todini, E., 2005, Flood forecasting using a fully distributed model: Application of the TOPKAPI model to the Upper Xixian Catchment: *Hydrology and Earth System Sciences*, v. 9, no. 4, p. 347-364.
- Lu, C. C., Chen, C. H., Yeh, T. C. J., Wu, C. M., and Yau, I. F., 2006, Integration of transfer function model and back propagation neural network for forecasting storm sewer flow in Taipei metropolis: *Stochastic Environmental Research and Risk Assessment*, v. 20, no. 1-2, p. 6-22.
- Lughofer, E., and Klement, E. P., FLEXFIS: A variant for incremental learning of Takagi-Sugeno fuzzy systems 2005, p. 915-920.
- Mamdani, E. H., and Assilian, S., 1975, An experiment in linguistic synthesis with a fuzzy logic controller: *International Journal of Man-Machine Studies*, v. 7, no. 1, p. 1-13.
- Martin Larsen, P., 1980, Industrial applications of fuzzy logic control: *International Journal of Man-Machine Studies*, v. 12, no. 1, p. 3-10.

- Mekong River Commission., 2011, Planning Atlas of the Lower Mekong River Basin Cambodia • Lao PDR • Thailand • Viet Nam.
- Minns, A. W., and Hall, M. J., 1996, Artificial neural networks as rainfall-runoff models: *Hydrological Sciences Journal*, v. 41, no. 3, p. 399-417.
- Moore, R. J., 2007, The PDM rainfall-runoff model: *Hydrology and Earth System Sciences*, v. 11, no. 1, p. 483-499.
- MRC, 2015a, http://ffw.mrcmekong.org/historical_rec.htm.
- , 2015b, <http://ffw.mrcmekong.org/south.htm>.
- Mujumdar, P. P., and Ghosh, S., 2008, Fuzzy logic-based approaches in water resource system modelling, *in* Singh, V. P., Abrahart, R. J., Anderson, M., Bengtsson, L., Cruise, J. F., Kothyari, U. C., See, L. M., Serrano, S. E., Solomatine, D. P., Stephenson, D., Strupczewski, W. G., and SpringerLink (Online service), eds., *Practical Hydroinformatics Computational Intelligence and Technological Developments in Water Applications*: Berlin, Heidelberg, Springer Berlin Heidelberg.
- Mukerji, A., Chatterjee, C., and Raghuwanshi, N. S., 2009, Flood Forecasting Using ANN, Neuro-Fuzzy, and Neuro-GA Models: *Journal of Hydrologic Engineering*, v. 14, no. 6, p. 647-652.
- Nayak, P. C., 2010, Explaining Internal Behavior in a Fuzzy If-Then Rule-Based Flood-Forecasting Model: *Journal of Hydrologic Engineering*, v. 15, no. 1, p. 20-28.
- Nayak, P. C., Sudheer, K. P., and Jain, S. K., 2007, Rainfall-runoff modeling through hybrid intelligent system: *Water Resources Research*, v. 43, no. 7.
- Nayak, P. C., Sudheer, K. P., and Ramasastri, K. S., 2005a, Fuzzy computing based rainfall-runoff model for real time flood forecasting: *Hydrological Processes*, v. 19, no. 4, p. 955-968.
- Nayak, P. C., Sudheer, K. P., Rangan, D. M., and Ramasastri, K. S., 2004, A neuro-fuzzy computing technique for modeling hydrological time series: *Journal of Hydrology*, v. 291, no. 1-2, p. 52-66.
- , 2005b, Short-term flood forecasting with a neurofuzzy model: *Water Resources Research*, v. 41, no. 4.

- Nguyen, N. N., 2012, Self evolving Takagi-Sugeno-Kang fuzzy neural network [Ph.D: Nanyang Technological University (NTU), 179 p.
- Nguyen, N. N., Zhou, W. J., and Quek, C., 2015, GSETSK: a generic self-evolving TSK fuzzy neural network with a novel Hebbian-based rule reduction approach: *Applied Soft Computing*, v. 35, p. 29-42.
- Nguyen, P. K. T., and Chua, L. H. C., 2012, The data-driven approach as an operational real-time flood forecasting model: *Hydrological Processes*, v. 26, no. 19, p. 2878-2893.
- Nguyen, P. K. T., Chua, L. H. C., and Son, L. H., 2014, Flood forecasting in large rivers with data-driven models: *Natural Hazards*, v. 71, no. 1, p. 767-784.
- Perfileva, I., Dubois, D., Prade, H., Esteva, F., Godo, L., and Hoďáková, P., 2012, Interpolation of fuzzy data: Analytical approach and overview: *Fuzzy Sets and Systems*, v. 192, p. 134-158.
- Price, R. K., and Vojinovic, Z., 2011, *Urban hydroinformatics : data, models and decision support for integrated urban water management*, London, IWA Publishing, Urban hydroinformatics series, 520 p. p.:
- Quah, K. H., and Quek, C., 2006, FITSK: online local learning with generic fuzzy input Takagi-Sugeno-Kang fuzzy framework for nonlinear system estimation: *IEEE TRANSACTIONS ON SYSTEMS, MAN, AND CYBERNETICS—PART B: CYBERNETICS*, v. 36, p. 166-178.
- Quah, K. H., and Quek, C., 2007, MCES: A novel Monte Carlo evaluative selection approach for objective feature selections: *Ieee Transactions on Neural Networks*, v. 18, no. 2, p. 431-448.
- Rath, S., Nayak, P. C., and Chatterjee, C., 2013, Hierarchical neurofuzzy model for real-time flood forecasting: *International Journal of River Basin Management*, v. 11, no. 3, p. 253-268.
- Refsgaard, J. C., 1996, CHAPTER 2: Terminology, modeling protocol and classification of hydrological model codes, *in* Abbott, M. B., and Refsgaard, J. C., eds., *Distributed Hydrological Modelling*: Dordrecht, Springer Netherlands.
- Remesan, R., Shamim, M. A., Han, D., and Mathew, J., ANFIS and NNARX based rainfall-runoff modeling, *in* *Proceedings Systems, Man and Cybernetics*,

2008. SMC 2008. IEEE International Conference on 12-15 Oct. 2008 2008, p. 1454-1459.
- Salas, J. D., Tabios, G. Q., and Bartolini, P., 1985, APPROACHES TO MULTIVARIATE MODELING OF WATER RESOURCES TIME SERIES1: JAWRA Journal of the American Water Resources Association, v. 21, no. 4, p. 683-708.
- Sene, K., 2010, Hydrometeorology. [electronic resource] : Forecasting and Applications, Dordrecht : Springer Science+Business Media B.V., 2010.
- Seo, D.-J., and Demargne, J., 2008, Use of Ensembles in Operational Hydrologic Forecasting in NWS, p. The 4th NCEP/NWS Ensemble User Workshop, Washington, 13–15 May 2008. http://www.emc.ncep.noaa.gov/gmb/ens/UserWkshop_May2008.html.
- Sherman, L. K., 1932, Streamflow from rainfall by the unit graph method: Eng. News Record v. 108:501-505.
- SMHI, 2013, <http://www.smhi.se/Professionella-tjanster/Professionella-tjanster/Miljo-och-klimat/Vattenmiljo/referensguide-till-smhi-vattenwebb-1.22742>.
- Solomatine, D., See, L. M., and Abrahart, R. J., 2008, Data-Driven Modelling: Concepts, Approaches and Experiences: Practical Hydroinformatics: Computational Intelligence and Technological Developments in Water Applications, v. 68, p. 17-30.
- Sudheer, K. P., Nayak, P.C., Rangan, D.M., 2000, Rainfall runoff modeling using artificial neural network technique.
- Sugeno, M., 1985a, Industrial applications of fuzzy control, Amsterdam ; New York, New York, N.Y., U.S.A., North-Holland ; Sole distributors for the U.S.A. and Canada, Elsevier Science Pub. Co., viii, 269 p. p.:
- Sugeno, M., 1985b, An Introductory Survey of Fuzzy Control: Information Sciences, v. 36, no. 1-2, p. 59-83.
- Takagi, T., and Sugeno, M., 1985, Fuzzy Identification of Systems and Its Applications to Modeling and Control: Ieee Transactions on Systems Man and Cybernetics, v. 15, no. 1, p. 116-132.

- Talei, A., 2013, Rainfall-runoff modelling with neuro-fuzzy systems [PhD: Nanyang Technological University, 246 p.
- Talei, A., and Chua, L. H. C., 2012, Influence of lag time on event-based rainfall-runoff modeling using the data driven approach: *Journal of Hydrology*, v. 438-439, p. 223-233.
- Talei, A., Chua, L. H. C., and Quek, C., 2010a, A novel application of a neuro-fuzzy computational technique in event-based rainfall-runoff modeling: *Expert Systems with Applications*, v. 37, no. 12, p. 7456-7468.
- Talei, A., Chua, L. H. C., Quek, C., and Jansson, P. E., 2013, Runoff forecasting using a Takagi-Sugeno neuro-fuzzy model with online learning: *Journal of Hydrology*, v. 488, p. 17-32.
- Talei, A., Chua, L. H. C., and Wong, T. S. W., 2010b, Evaluation of rainfall and discharge inputs used by Adaptive Network-based Fuzzy Inference Systems (ANFIS) in rainfall-runoff modeling: *Journal of Hydrology*, v. 391, no. 3-4, p. 248-262.
- Tikk, D., Csaba Johanyák, Z., Kovács, S., and Wong, K. W., 2011, Fuzzy rule interpolation and extrapolation techniques: Criteria and evaluation guidelines: *Journal of Advanced Computational Intelligence and Intelligent Informatics*, v. 15, no. 3, p. 254-263.
- Ting, C. W., and Quek, C., 2009, A novel blood glucose regulation using TSK0-FCMAC: A fuzzy CMAC based on the zero-ordered TSK fuzzy inference scheme: *IEEE Transactions on Neural Networks*, v. 20, no. 5, p. 856-871.
- Tsukamoto, Y., 1979, An approach to fuzzy reasoning method, *in* Gupta, M. M., Ragade, R. R., and Yager, R. R., eds., *Advances in Fuzzy Set Theory and Applications*: North-Holland, Amsterdam, p. 137–149.
- Tung, W. L., and Quek, C., 2010, eFSM-A Novel Online Neural-Fuzzy Semantic Memory Model: *Ieee Transactions on Neural Networks*, v. 21, no. 1, p. 136-157.
- Turksen, I. B., and Zhong, Z., 1988, An Approximate Analogical Reasoning Approach Based on Similarity Measures: *Ieee Transactions on Systems Man and Cybernetics*, v. 18, no. 6, p. 1049-1056.

- Wang, W.-C., Chau, K.-W., Cheng, C.-T., and Qiu, L., 2009, A comparison of performance of several artificial intelligence methods for forecasting monthly discharge time series: *Journal of Hydrology*, v. 374, no. 3–4, p. 294-306.
- Whitlock, J. R., Heynen, A. J., Shuler, M. G., and Bear, M. F., 2006, Learning induces long-term potentiation in the hippocampus: *Science*, v. 313, no. 5790, p. 1093-1097.
- Wu, S. Q., and Er, M. J., 2000, Dynamic fuzzy neural networks - A novel approach to function approximation: *Ieee Transactions on Systems Man and Cybernetics Part B-Cybernetics*, v. 30, no. 2, p. 358-364.
- Yang, L. Z., and Shen, Q., 2013, Closed form fuzzy interpolation: *Fuzzy Sets and Systems*, v. 225, p. 1-22.
- Yen, J., and Langari, R., 1999, *Fuzzy logic : intelligence, control, and information*, Upper Saddle River, N.J. : Prentice Hall, c1999.
- Zadeh, L. A., 1965, Fuzzy Sets: *Information and Control*, v. 8, no. 3, p. 338-&.
- , 1973, Outline of a New Approach to the Analysis of Complex Systems and Decision Processes: *IEEE Transactions on Systems, Man, and Cybernetics*, v. SMC-3, no. 1, p. 28-44.
- , 1994, *Fuzzy-Logic, Neural Networks, and Soft Computing: Communications of the Acm*, v. 37, no. 3, p. 77-84.
- Zhang, C., Wang, Y. L., Zhang, L. L., and Zhou, H. C., 2012, A fuzzy inference method based on association rule analysis with application to river flood forecasting: *Water Science and Technology*, v. 66, no. 10, p. 2090-2098.