

الجمهورية الجزائرية الديمقراطية الشعبية

PEOPLE'S DEMOCRATIC REPUBLIC OF ALGERIA

وزارة التعليم العالي والبحث العلمي

Ministry of Higher Education and Scientific Research

جامعة أبي بكر بلقايد - تلمسان

Aboubakr Belkaïd University – Tlemcen –

Faculty of TECHNOLOGIE



THESIS

Presented for obtaining the rank of **DOCTOR OF SCIENCES**

In: Hydraulic

Speciality: Hydraulic

By: ZAKHROUF MOUSAAB

Subject

**Development of a Neuronal System for Modeling Streamflows of
Some Watersheds in Algeria.**

Sustained publicly, on the 28/ 03/2021 , in front of a jury composed of:

Mr. BENMANSOUR Abdelhalim	Professeur	Univ. Tlemcen	Président
Mr. BOUCHELKIA Hamid	Professeur	Univ. Tlemcen	Directeur de thèse
Mr. STAMBOUL Madani	Professeur	Univ. Laghouat	Co- Directeur de thèse
Mr. REMINI Boualem	Professeur	Univ. Blida	Examineur 1
Mr. BERREKSI Ali	MCA	Univ. Bejaia	Examineur 2

الجمهورية الجزائرية الديمقراطية الشعبية

REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE

وزارة التعليم العالي والبحث العلمي

Ministère de l'Enseignement Supérieur et de la Recherche Scientifique

جامعة أبي بكر بلقايد - تلمسان

Université Aboubakr Belkaïd – Tlemcen –

Faculté de TECHNOLOGIE



THESE

Présentée pour l'obtention du **grade** de **DOCTEUR EN SCIENCES**

En : Hydraulique

Spécialité : Hydraulique

Par : ZAKHROUF MOUSAAB

Sujet

Développement d'un Système Neuronale Pour La Modélisation Des Débits De Quelques Bassins Versants en Algérie

Soutenue publiquement, le 28 / 03 / 2021 , devant le jury composé de :

Mr. BENMANSOUR Abdelhalim	Professeur	Univ. Tlemcen	Président
Mr. BOUCHELKIA Hamid	Professeur	Univ. Tlemcen	Directeur de thèse
Mr. STAMBOUL Madani	Professeur	Univ. Laghouat	Co- Directeur de thèse
Mr. REMINI Boualem	Professeur	Univ. Blida	Examineur 1
Mr. BERREKSI Ali	MCA	Univ. Bejaia	Examineur 2

Abstract

The main purpose of this research is to develop highly efficient, daily streamflow time series models based on the artificial neural networks approaches for five watersheds located in different geographical and Hydro-Climatic regions in Algeria.

Different type of artificial neural networks including feed forward neural networks (FFNN), adaptive neuro-fuzzy inference system (ANFIS), hybrid wavelet transformation-neural networks, and recurrent neural networks models was be suggested. Also applying different methods like genetic algorithm, particle swarm optimisation and k-fold cross validation to enhance the performance of artificial neural network models.

The results obtained showed that the time series models based artificial neural networks are found to be very promising alternative to modelling and forecasting of short term (one-step ahead) and long term (multi-step ahead) streamflow.

Key words : Time series models, Daily streamflow forecasting, Artificial neural networks, Watersheds, Algeria.

ملخص

الغرض الرئيسي من هذا البحث هو تطوير نماذج سلاسل زمنية ذات كفاءة عالية للتنبؤ بالتدفق اليومي لخمسة مستجمعات مياه تقع في مناطق جغرافية ومناخية مختلفة في الجزائر ، تعتمد على نهج الشبكات العصبية الاصطناعية.

تم اقتراح أنواع مختلفة من الشبكات العصبية الاصطناعية بما في ذلك الشبكات العصبية ذات التغذية الأمامية ، ونظام الاستدلال العصبي الضبابي التكيفي ، والشبكات العصبية الهجينة مع التحويل باستعمال الموجات ، ونماذج الشبكات العصبية المتكررة. أيضًا تم تطبيق طرق مختلفة لتعزيز أداء نماذج الشبكات العصبية مثل الخوارزمية الجينية ، تحسين سرب الجسيمات والتحقق المتقاطع. أظهرت النتائج التي تم الحصول عليها أن الشبكات العصبية الاصطناعية القائمة على نماذج السلاسل الزمنية تعد بديلاً واعدًا جدًا للنمذجة والتنبؤ بالتدفق على المديين القصير و الطويل.

الكلمات المفتاحية . نماذج سلاسل زمنية ، التنبؤ بالتدفق اليومي، الشبكات العصبية الاصطناعية، مستجمعات مياه، الجزائر.

Résumé

L'objectif principal de cette recherche est de développer des modèles efficaces de séries chronologiques de débit journalier, basés sur les approches de réseaux de neurones artificiels appliqué sur cinq bassins versants situés dans différentes régions géographiques et hydro-climatiques en Algérie.

Différents types de réseaux de neurones artificiels ont été proposés, notamment les réseaux de neurones de type feed forward (FFNN), le système adaptatif d'inférence neuro-floue (ANFIS), les réseaux de neurones hybrides avec la transformation en ondelettes et les modèles de réseaux de neurones récurrents. L'application de différentes méthodes telles que l'algorithme génétique, l'optimisation des essaims de particules et la validation croisée k-folds a également été proposée pour améliorer les performances des modèles de réseaux de neurones artificiels.

Les résultats obtenus ont montré que les modèles de séries chronologiques basés sur l'approche de réseaux de neurones artificiels s'avèrent être une alternative très prometteuse à la modélisation et à la prévision des débits à court terme (un pas en avant) et à long terme (plusieurs pas en avant).

Mots clés : Modèles des séries chronologiques, Prévision du débit journalier, Réseaux de neurones artificiels, Bassins versants, Algérie.

Acknowledgements

I would like to take this opportunity to express my deepest appreciation to my supervisor, Dr Hamid Bouchelkia, for his continued support and motivation during this period of this project. My gratitude is also extended to my co-supervisor , Dr. Madani Stamboul, for his immense experience, useful encouragement and advice.

I would like to give my special thanks to all of: Prof. Benmansour Abdelhalim, Abou Bekr Belkaid university, Algeria ; Prof. Sungwon Kim, Dongyang University, south Korea ; and Prof. Salim Heddami, University 20 aout 1955, skikda, Algeria.

Finally, I would like to thank my family members, and all my friends that supported me to finish this work.

Moussaab Zakhrouf

List of publication

Zakhrouf, M., Bouchelkia, H., & Stamboul, M. (2015). Neuro-wavelet (WNN) and neuro-fuzzy (ANFIS) systems for modeling hydrological time series in arid areas. A case study: The catchment of Ain Hadjadj (Algeria). *Desalination and Water Treatment*, 57, 17182–17194.

Zakhrouf, M., Bouchelkia, H., Stamboul, M., Kim, S., and Heddami, S. (2018). Time series forecasting of river flow using an integrated approach of wavelet multi-resolution analysis and evolutionary data-driven models. A case study: Sebaou River (Algeria). *Physical Geography*, Vol. 39, No. 6, pp. 506-522.

Zakhrouf, M., Bouchelkia, H., Kim, S., Stamboul, M. (2020). Novel insights for streamflow forecasting based on deep learning models combined the evolutionary optimization algorithm. (under review).

Zakhrouf, M., Bouchelkia, H., Stamboul, M., Kim, S., Singh, V. (2020). Implementation on the evolutionary machine learning approaches for streamflow forecasting: case study in the Seybous River, Algeria. *J. Korea Water Resour. Assoc.*, 53, 395-408.

Zakhrouf, M., Bouchelkia, H., Stamboul, M., Kim, S. (2020). Novel hybrid approaches based on the evolutionary strategy for streamflow forecasting in the Chellif River, Algeria, *Acta Geophysica*, 68, 167-180.

Table Of Contents

Abstract.....	i
Acknowledgements	iii
List of publication	iv
Table Of Contents	viii
List of figures.....	x
List of tabels.....	xi
List of abriviations	xii
General Introduction.....	1
Chapter I : Artificial Neural Networks (ANN) Overview	8
I.1. Introduction.....	9
I.2. Basic Neural Network Components.....	9
I.3. Some types of neural networks	10
I.3.1. Feed Forward Neural Networks	10
I.3.2. Recurent Neural Networks	13
I.3.2. 1. Elman RNNs (ERNN)	13
I.3.2. 2. Long Short Terme Memory (LSTM)	15
I.3.2. 3. Gated Recurrent Unit (GRU) Network.....	16
I.3.3. Adaptive neuro-fuzzy inference system (ANFIS)	18
I.3.4. Wavelet-based neural networks (WNN)	20
I.3.4. 1. Wavelet transform (WT).....	20
I.3.4. 2. Discrete wavelet transform (DWT)	22
I.4. Conclusion	24
Chapter II : Applied Neuro-Wavelet and Neuro-Fuzzy systems on the catchment of Ain	
Safra.....	25
II.2. Introduction	26
II.2. Materials and Methods.....	28
II.3. Study area and data used	28

II.4. Input selection.....	30
II.5. Implementation of ANN and WNN	32
II.6. Implementation of ANFIS	34
II.7. Performance criteria.....	36
II.8. Results and discussion.....	37
II.9. Conclusion.....	44
Chapter III : Applied Evolutionary Wavelet-ANFIS And Wavelet- FFNN On The Sebaou River	46
III.1. Introduction	47
III.2. Study area and data.....	49
III.3. Materials and Methods.....	51
III.3.1. Genetic algorithm	51
III.4. Methodology	53
III.5. Results and application	59
III.6. Conclusion.....	66
Chapter IV : Applied Evolutionary Recurent Neural Networks. A case study: Chellif and Soummam Watersheds	68
IV.1. Introduction.....	69
IV. 2. Materials and Methods	71
IV.3. Parameters and Hyperparameters Tunning.....	72
IV.3.1 Adaptive moment estimation (ADAM) algorithm.....	72
IV.3.2. Particle swarm optimization (PSO) optimization algorithm	73
IV.4. Research area and data analysis	74
IV.4.1. Soummam and Chellif River Watersheds	74
IV.4.2. ACF and PACF analysis.....	76
IV.5. Statistical Indices.....	80

IV.6. Implementation of developed models	81
IV.7. Results and Disussion.....	84
IV.7. 1. Sidi Aich station.....	84
IV.7. 2. Ponteba Defluent station	87
IV.8. Conclusion	90
Chapter V : Evolutionary Neuro-Wavelet and Neuro-Fuzzy systems for multi step ahead forecasting : case study in the Seybous River.....	92
V.1. Introduction	93
V.2. Materials and Methods.....	96
V.2.1. K-fold cross validation (CV).....	96
V.3. Study area and data description.....	97
V.4. Methodology	98
V.4.1 Models development.....	98
V.4.2 Coding for the FFNN, ANFIS, and WFFNN models	99
V.5. Measures of accuracy.....	101
V.6. Results and Discussion	102
V.7. Conclusion.....	108
Chapter VI : Applied Evolutionary Neuro-Wavelet by three diffrents strategies for multi step ahead forecasting. Case study : Chellif River.....	110
V.1. Introduction	111
VI.2. Materials and Methods	113
VI.3. Methodology description	113
VI.4. FFNNs-GA and WFFNNs-GA models	114
VI.5. Study area and data.....	117
VI.6. Measures of accuracy	118
VI.7.Results and discussion.....	119
VI.8. Conclusion.....	130
General Conclusion	131

References.....136

List of figures

Figure. I.1. McCulloch and Pitts formal neuron structure.....	9
Figure. I.2. Conventional Feed Forward Neural Networks (FFNN) architecture.....	12
Figure.I.3. Recurrent Neural Networks processing.....	13
Figure. I.4. Elman Recurrent Neural Networks.....	14
Figure. I.5. Long-short term memory (LSTM) gates processing.....	15
Figure.I.6. Gated recurrent units processing.....	17
Figure. I.7. Operation principle of the ANFIS network based on the fuzzy inference system, Takagi–Sugeno (TS) type.....	20
Figure.I.8. Wavelet transform mechanism.....	21
Figure.I.9. The schematic diagram of WNN model.....	23
Figure. II.1. Ain Safra watershed (Derdour et al).....	32
Figure. II.2. Average daily streamflow and cumulative rainfall measured at stations: Ain Hadjadj - Ain safra, (period 1973/1999).....	33
Figure.II.3. (a) Simple Correlogram of rainfall, (b) density variance spectrum of rainfall....	34
Figure.II.4. (a) Simple Correlogram of streamflow, (b) density variance spectrum of streamflow.....	35
Figure.II.5. The optimal number of neurons for FFNN model in the testing phase.....	36
Figure.II.6. The optimal number of neurons for WFFNN model in the testing phase.....	37
Figure.II.7. The optimal number of MFs for ANFIS model in the testing.....	39
Figure.II.8. Plot of (a). Observed and simulated hydrographs, (b). Error plots along the magnitude of streamflow, (c). Scatter of observed and simulated streamflow for WFFNN model during testing phase.....	42
Figure.II.9. Plot of (a). Observed and simulated hydrographs, (b). Error plots along the magnitude of river streamflow, (c). Scatter of observed and simulated streamflow for ANFIS model during testing phase.....	43
Figure.II.10. Plot of (a). Observed and simulated hydrographs, (b). Error plots along the magnitude of streamflow, (c). Scatter of observed and simulated streamflow for FFNN model during testing phase.....	44
Figure.II.11. Peak flow estimate and relative peak error for WFFNN model during testing phase.....	45
Figure.II.12. Peak flow estimate and relative peak error for ANFIS model during testing phase.....	46
Figure.II.13. Peak flow estimate and relative peak error for FFNN model during testing phase.....	47
Figure. III.1. Sebaou watershed (et al).....	55
Figure. III.2. Average daily streamflow measured at Baghlia gaging station.....	56
Figure. III.3. The flowchart of the optimization process of GA.....	60
Figure III.4. Chromosome encoding of WFFNN.....	62
Figure.III.5. Chromosome encoding of WANFIS.....	63
Figure. III.6. Hydrograph and scatter plot in the testing phase (a) WANFIS, (b) ANFIS.....	68
Figure. III.7. Hydrograph and scatter plot in the testing phase (a) WFFNN, (b) FFNN.....	69
Figure. III.8. Error and relative error in the testing phase (a) WANFIS, (b) ANFIS.....	70

Figure. III.9. Error and relative error in the testing phase (a) WFFNN, (b) FFNN.	71
Figure. IV.1. Soummam watershed (Allili-Ailane et al, 2015).....	84
Figure. IV.2. Chellif watershed (Benhattab et al, 2011)	84
Figure. IV.3. Observed streamflow hydrograph (6years) of Ponteba Defluent station.....	85
Figure. IV.4. Observed streamflow hydrograph (6years) of Sidi Aich station.....	86
Figure. IV.5. a) Autocorrelation and b) partial autocorelation correlograms for Ponteba Defluent streamflow time series.....	87
Figure. IV.6. a) Autocorrelation and b) partial autocorelation correlograms for Sidi Aich streamflow time series.....	88
Figure. IV.7. The flow chart on applying the models in forecasting streamflow.	91
Figure. IV.8. Observed and simulated hydrographs and scater plots during testing phase for Sidi Aich station : a) FFNN model, b) RELM model, c) ANFIS model, d) GRU model.	96
Figure. IV.9. Observed and simulated hydrographs and scater plots during testing phase for Ponteba Defluent station : a) FFNN model, b) ELM model, c) LSTM model, d) GRU model.	99
Figure. V.1. Representation of k-fold cross validation method.....	110
Figure. V.2. Observed streamflow hydrograph (14years)	111
Figure. V.3. Chromosome encoding for FFNNs model	113
Figure.V.4. Chromosome encoding for ANFIS model	114
Figure. V.5. Chromosome encoding for WFFNNs model	115
Figure. V.6. Scatter diagram for FFNNs, ANFIS and WFFNNs models (testing phase)	121
Figure. V.7. Relative errors of the peak flows for FFNNs, ANFIS and WFFNNs models (testing phase).....	122
Figure. VI.1. Conventional FFNNs architecture (a) Independent method, (b) Joint method.....	133
Figure. VI.2. Chromosome encoding for WFFNNs model	135
Figure. VI.3. Observed streamflow hydrograph (14years).....	136
Figure. VI.4. Scatter diagram for WFFNNs models using MISO, MIMO, and MISMOstrategies (Testing phase)	146
Figure. VI.5. Scatter diagram for FFNNs models using MISO, MIMO, and MISMOstrategies (Testing phase).....	148

List of tables

Table. I.1. Most commonly used activation functions.	11
Table. II.1. Results obtained by the models: WFFNN, ANFIS and FFNN.	41
Table. III.1. Optimal structure of FFNN and WFFNN obtained by GA.	65
Table. III.2. Optimal structure of ANFIS and WANFIS obtained by GA.	66
Table. III.3. Comparison between the performance results obtained by the models: WANFIS, ANFIS, WFFNN, and FFNN in the testing phase... ..	67
Table. IV.1. Model based inputs propositions.	89
Table. IV.2. Variables encoded and their default domains.	93
Table. IV.3. The optimal structure for FFNN, ELM, LSTM and GRU models using PSO for Sidi Aich station.	94
Table. IV.5. Comparison between the performance results obtained by the: FFNN, ELM, LSTM and GRU models in the training and testing phases for Sidi Aich and Ponteba Defluent stations.	95
Table. IV.4. The optimal structure for FFNN, ELM, LSTM and GRU models using PSO for Ponteba Defluent station.	97
Table. V.1. The optimal structure for FFNNs model using GA.	117
Table. V.2. The optimal structure for ANFIS model using GA.	118
Table. V.3. The optimal structure for WNNs model using GA.	119
Table. V.4. Comparison between the performance results obtained by the models:	120
Table. VI.1. The optimal structure for WFFNNs model based MISO strategy using GA.	139
Table. VI.2. The optimal structure for WFFNNs model based MISMO strategy using GA.	140
Table. VI.3. The optimal structure for WFFNNs model based MIMO strategy using GA.	141
Table. VI.4. Comparison between the performance results obtained by the MISO, MIMO, and MISMO strategies in the training and testing phases for WFFNNs models.	143
Table. VI.5. Comparison between the performance results obtained by the MISO, MIMO and MISMO strategies in the training and testing phases for FFNNs models.	144

List of abbreviations:

AC	Ant colony
ACF	Autocorrelation function
ADAM	Adaptive moment estimation
AFHL	Activation function type in hidden layers
AFOL	Activation function type in output layer
ANFIS	Adaptive neuro-fuzzy inference system
ARMA	Auto regressive moving average
ARIMA	Auto regressive integrated moving average
ARIMAX	Auto Regressive Integrated Moving Average with exogenous input
Bior	Biorthogonal mother wavelette
BP	Backpropagation
BPTT	Backpropation through time
BS	Batch size
Coif	Coiflets mother wavelette
CV	Cross validation
CVE	Cross validation error
CWT	Continuous wavelet transform
D	Input delay
Db	Daubechies mother wavelette ,
DRT	Definition of if-then (AND operation/OR operation) rules type
Dsgmf	Built-in Gaussian membership function
DWT	Discrete wavelet transform
EAs	Evolutionary algorithms
ELM	Extreme learning machines
EP ()	Evolutionary programming
ERNN	Elman rnns
ES	Evolutionary strategy
FIS	Fuzzy inference systems
FFNN	Feed forward neural networks
FOM	First-order momentum
FSR	Firing strength of a rule
GA	Genetic algorithm
Gaussmf	Gaussian curve membership function
GEP	Gene expression programming
GP	Genetic programming
GRU	Gated recurrent unit
Har	Haar mother wavelette
IWB	Initial connection weights and bias coefficients ().
Logsig	Log-sigmoid transfer functions

LMBP	Levenberg–marquardt backpropagation
LR	Linear regression
lr	Larning rate
LSTM	Long short terme memory
MARS	Multivariate adaptive regression spline
Max	Maximum
MA	Moving average
MESA	Maximum entropy spectral analysis
MFs	Membership functions
MIMO	Multi-input multi-output
Min	Minimum
MISO	Multi-input single-output
MISMO	Multi-input several multi-output
MSE	Mean squared error
MT	Model tree
NHL	Number of hiden layers
NNHL	Number of neurons in hidden layers
NMF	Number of membership functions
NSE	Nash-Sutcliffe efficiency coefficient
OOS-CV	Out-of-sample cross validation
PACF	Partial autocorrelation function
PFC	Peak flow criteria
<i>Pimf</i> Π-	Shaped membership function
Probor	Probabilisticor
Prod	Product
PSO	Particle swarm optimization
Purelin,	Linear transfer functions
R^2	Determination coefficient
R	Correlation coefficient
ReLU	Rectified linear unit transfer functions
RMSE	Root mean square error
RNN	Recurent neural networks
SNR	Signal-to-noise ratio
Satlin,	Symmetric saturating linear transfer functions
Sigmoid	Sigmoid transfer function
SOM	Second-order momentum
SVM	Support vector machines
Sym	Symlets mother wavelette
Tansig	Hyperbolic tangent sigmoid transfer functions
TMF	Type of membership functions
Trapmf	Trapezoidal-shaped membership function

Trimf	Triangular-shaped membership function
TS	Takagi–sugeno
WANFIS	Wavelet adaptive neuro fuzzy inference system
WFFNN	Wavelet based feed forward neural networks model
WMRA	Wavelet multi-resolution analysis
WNN	Wavelet neural networks
WT	Wavelet transform

General introduction

1. Overview

One of the declarations of the Rio Conference (June 1992) states:

"The comprehensive management of freshwater as a finite and fragile resource and the integration of sectoral water plans and programmes into the framework of national economic and social policies are of paramount importance for interventions in the 1990s and beyond.

Efficient and sustainable water management is not limited to guaranteeing, on average, sufficient quantity and quality for human demands (drinking water, industrial water, irrigation water, etc.) and for the needs of natural environments; it must also take into account the occurrence of extreme events, such as low water levels and floods (Riad, 2004).

Considering the waters of rivers and valleys as one of the available water resources, so accurate forecasting of these resources is considered a key element in drought analysis and the design of water related infrastructures and the management of dam stocks. There is ongoing research to enhance the accuracy and reliability of river flow forecasting. Many types of river flow forecasting models was developed and investigate to better manage scarce water resources and minimize the risk of potential floods.

2. Motivation

Algeria, is one of the southern countries of the Mediterranean basin that suffers from water scarcity from one season to another, and from one year to another. The water potential is globally estimated at 19 billion m³/year (corresponding to about 600 m³/inhab/year) and Algeria is in the category of water-poor countries with regard to the scarcity threshold set by the World Bank at 1000 m³/inhab/year (UN 2002). Water flows are characterised by significant seasonal and interannual irregularity, violence and rapidity of floods. The climate in Algeria is semi-arid (200mm to 500mm) from which resources are increasingly limited and difficult to exploit. (Kettab et al. 2004).

Modelling approaches are needed to assess and compare water resources to changes in their use and thus define the current and future capacity of resources to meet demands. Modelling also allows analysing management scenarios and measuring possible trends (population growth, climate change, etc.) with the implementation of flexible policies and measures, regularly assessed and corrected (Droogers and Aerts quoted by Droogers et al. 2011). Decision-makers need such methods to assess whether future water needs can be met and to define the most appropriate adaptation strategies to meet demands and prevent tensions in use.

3. Problematic and thesis objective

The transformation of rainfall into streamflow results from a number of complex mechanisms that occur simultaneously at different spatial and temporal scales.

The performance of a flow prediction system depends on many factors, but first and foremost on the adequacy of the technology used and the most influential physical processes of the forecasting context (Fredric).

Physical models generally involve solving a system of differential equations to model the different components of the hydrological cycle in the catchment area.

However, although these physically based models help to understand the underlying physics of hydrological processes, their application is limited due to the large amount of input data and computation time required (chouaib, 2016).

On the other hand, time-series models rely primarily on measured data and represent the relationship between inputs (which represent the measured phenomenon values in the past) and outputs (which represent the measured phenomenon value in the present) without taking into account the complex nature of the underlying process.

For this reason, the main purpose of this research is to develop highly efficient, reliable and accurate daily streamflow time series models based on the artificial neural networks approaches for five catchments located in different geographical and Hydro-Climatic regions in Algeria.

In this study different type of artificial neural networks including feed forward neural networks (FFNN), adaptive neuro-fuzzy inference system (ANFIS), hybrid wavelet-FFNN and wavelet-ANFIS models will be developed. Also applying different methods like genetic algorithm, particle swarm optimisation and k-fold cross validation to enhance the performance of artificial neural network models.

We have considered each watershed as a separate case study than the other and the models and methods applied for each case study are different from the other in order to vary the methods to know the most efficient.

4. Structure of the thesis

This thesis contains a total of eight chapters divided on five key sections of the General introduction, Material, Short term forecasting, Long term forecasting and General conclusion.

Section 1, General introduction

- This introduction reflects on the issue and the key motives for undertaking this study. This highlights the main goals of the research and summarizes the topic.

Section 2, Material

- Chapter One- Artificial neural networks, describes the theoretical background of artificial neural networks and some artificial neural networks types are described briefly including feed forward neural network (FFNN), Elmen network (ELN), long short terme memory (LSTM), gated recurent unit (GRU), adaptive neuro-fuzzy system (ANFIS) and wavelet neural networks (W-FFNN, W-ANFIS).

Section 3, Short term forecasting

- Chapter Two- This chapter presents the application of artificial neural networks models (FFNN, ANFIS, W-FFNN) on the catchment of Ain Safra to forecast one day ahead.
- Chapter Three - In this chapter, application of wavelet neural networks (wavelet-FFNN, wavelet-ANFIS) were developed to forecast one day ahead of Sebaou river flow. The results of this hybrid models were compared with convonsional models (FFNN, ANFIS). Also an algorithme genetic was used to optimise the architecture of our models.
- Chapter Four - This chapter investigates the application of recurent neural networks (ELN, LSTM and GRU) for one day ahead of Soummam and Chellif watersheds. The results were compared with simple FFNN model. The best parameters of our models were selectioned using particle swarm optimisation algorithm.

Section 4, Long term forecasting

- Chapter Five- In this chapter the application of multi step ahead daily river flow forecasting is used by applying FFNN, ANFIS and wavelet-FFNN models. The selectioned models was applied on the Seybouss river flow to forecast three days ahead in the futur. A novel approach based on the using of genetic algorithm with the k-folds cross validation method was investigated to enhance the performance of our models.
- Chapter Six - In this chapter the application of wavelet-FFNN and standar FFNN are investigated for various lead-times of 1 to 4 days ahead for the Chellif river based on three evolutionary strategies [i.e., multi-input multi-output (MIMO), multi-input single-output (MISO), and multi-input several multi-output (MISMO)]. As we done in

the chapter seven, the models were improved using the hybridation of genetic algorithm with k-fold cross validation to find the best parameters of our models.

Section 5, General conclusion

- Summary of research outcomes, review of study results and the generally result are presented in this chapter.

Chapter. I

Artificial Neural Networks Overview.

I.1. Introduction

Artificial neural network is considered as one of the modern mathematical computational methods which are used to solve non-linear and dynamic problems, where research in the field of neural networks are attracting increasing attention in recent years, And that's because it has that ability to learn and to generalise data, by using the parallel processing nervous units.

This chapter briefly describes the mechanism of the formal neuron. In addition to the most common neural networks including the feed forward network, Elman network, long short term memory, gated recurrent unit, adaptive neuro-fuzzy system and wavelet neural network.

I.2. Basic Neural Network Components

ANN is a parallel computational system based on the architectural and functional standard of biological networks (Imrie et al, 2000). Across the various neural network models that have been developed over the years, all of them share a specific building block known as a neuron. The most commonly used neuron model is based on the work of McCulloch and Pitts (1943) and is shown in Figure I.1.

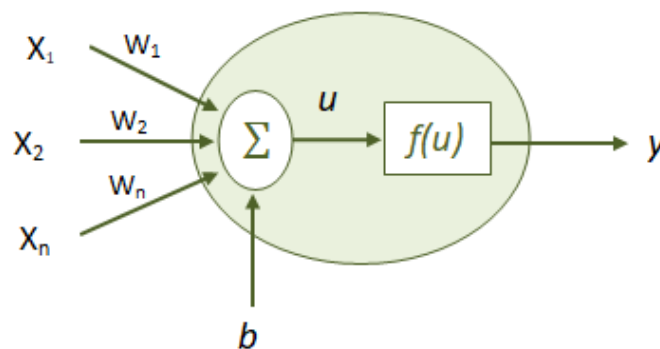


Figure. I.1. McCulloch and Pitts formal neuron structure.

In Figure (I.1), each neuron consists of two parts: the net function and the activation function. The net function determines how the network inputs are combined inside the neuron. In this figure, a weighted linear combination is adopted:

$$u = w^T x + b \quad (\text{I.1})$$

where: u represents the weighted sum of the inputs of the neuron, x represents the input connected to the neuron, w denotes the weight of the connection connecting the input to the neuron, and b the internal threshold of the neuron.

The output of the neuron, denoted by y in this figure, is related to the network input u via a linear or nonlinear transformation called the activation function:

$$y = f(u) \quad (\text{I.2})$$

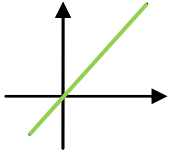
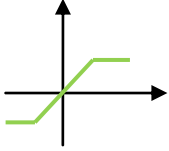
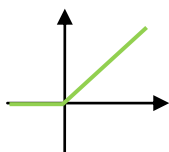
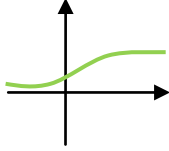
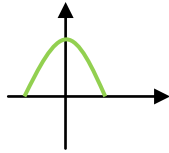
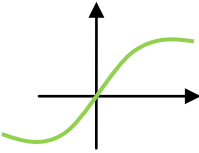
In various neural network models, different activation functions have been proposed. The most commonly used activation functions are summarized in Table (I.1).

I.3. Some types of neural networks

I.3.1. Feed Forward Neural Networks

Artificial neural network can be seen as a mathematical model of distributed processing, composed of several non-linear computational elements (neurons), operating in parallel and connected to each other by weights. Each elementary processor calculates a single output based on the information it receives. Through their parallel processing of information and their mechanisms inspired by nerve cells (neurons), they infer emergent properties to solve problems once described as complex.

Table. I.1. Most commonly used activation functions.

Function name	Mathematic forme	Geometric forme
Linear or Purline	$f = n$	
Satlins	$f = \begin{cases} -1, & \text{if } n \leq -1 \\ n, & \text{if } -1 \leq n \leq 1 \\ 1, & \text{if } 1 \leq n \end{cases}$	
ReLue	$f = \begin{cases} n & \text{if } n > 0 \\ 0 & \text{if } n \leq 0 \end{cases}$	
Sigmoïde or Logsig	$f = \alpha \frac{e^{kn} - 1}{e^{kn} + 1}$	
Radbas	$f = e^{-n^2/\beta^2}$	
Tansig	$f = \frac{1 - e^{-2n}}{1 + e^{2n}}$	

In the case of feed forward multi layer network, neurons are organized in layers. The input information is transmitted by successive layers, to finally obtain the output result. In this topology, neurons are divided into three classes: input neurons, hidden neurons and output neurons. Each neuron in a layer is connected to all the neurons of the previous layer and of the next layer (except for the input and output layers) Fig(I.2).

If we consider a multi-layer model with N input neurons, activated by an input vector x (of size N), and by $w_{ij}^{0,1}$ the weight corresponding to the connection between neuron i of layer 1 and neuron j of layer 2, the output h_{1j} of each of the neurons of the first hidden layer will be expressed by :

$$O_j^1 = g_j(b_j^1 + \sum_{i=1}^N w_{ij}^{0,1} x_i) \quad (\text{I.3})$$

Où g_j est la fonction d'activation décrite précédemment, et b_j^1 est un paramètre supplémentaire appelé biais, et dont son rôle est de rajouter un degré de liberté supplémentaire en agissant sur la position de la frontière de décision.

This same process expressed by equation (1) can be repeated for the other layers.

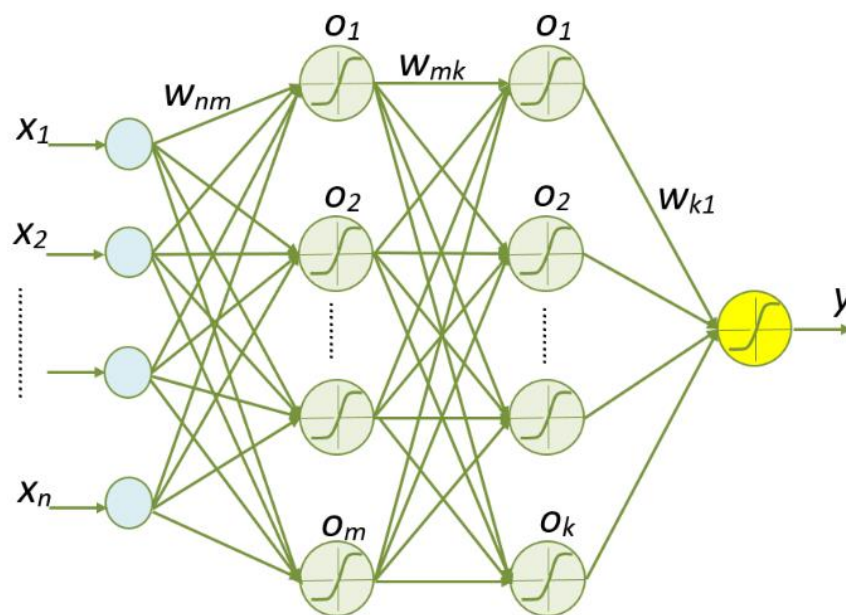


Figure. I.2. Conventional Feed Forward Neural Networks (FFNN) architecture.

I.3.2. Recurrent Neural Networks

The recurrent neural network (RNN) is designed to detect the sequential actions of the data set in order to anticipate the next likely scenario. It is an efficient approach to the analysis of sequential data, such as time series data.

In the traditional neural network (Feed forward) we assume that all inputs and outputs are independent of each other, but in cases where it is necessary to forecast the next value of a time series chain, the previous value is required and needs to be remembered by the model. So, the idea behind the RNN is to make the model remember through the time, this memory which captures information about what has been calculated so far Fig(I.3).

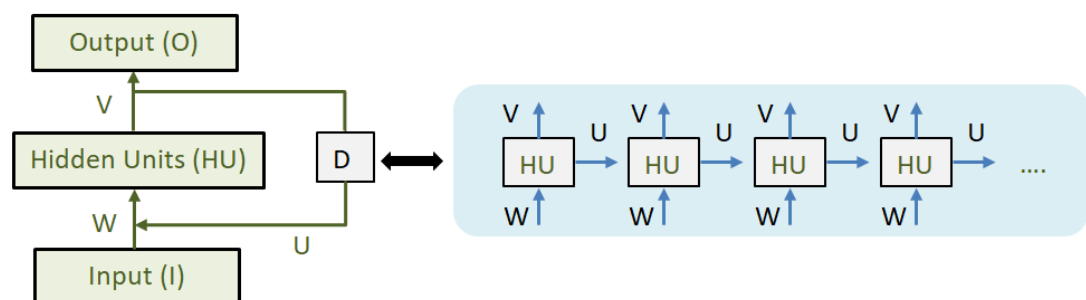


Figure.I.3. Recurrent Neural Networks processing.

I.3.2. 1. Elman RNNs (ERNN)

Elmann Recurrent Neural Networks (ELM) has been suggested by (Elman 1990) to generalize feedforward neural networks in order to help handle organized data sequences including time-series.

The fundamental configuration of the Elman Neural Network model is the input layer, the hidden layer, the output layer and the context layer.

The context layer is a feedback connection from the hidden layer to the input layer Fig (I.4). Connections among the input layer, the hidden layer and the output layer can be considered as a feed-forward network, this part is similar to the traditional feed forward network.

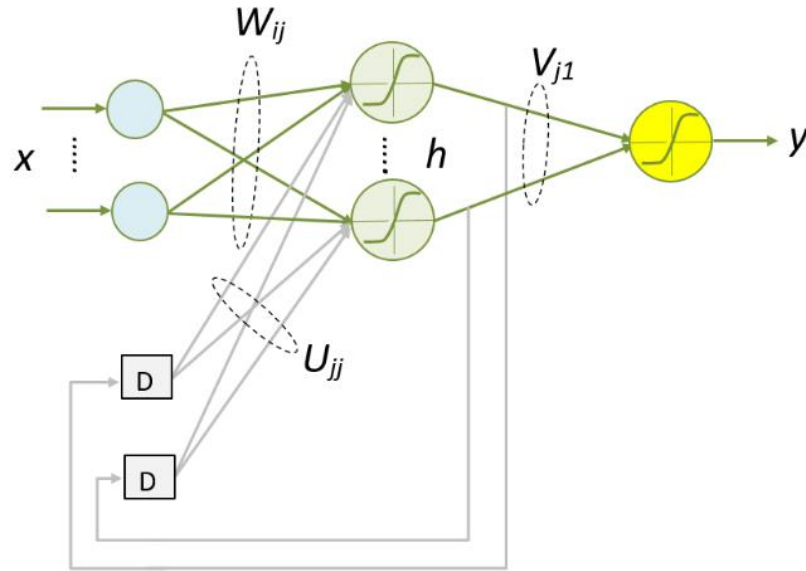


Figure. I.4. Elman Recurrent Neural Networks.

The feed back of the output of hidden layer to input layer named the context layer. After Elman the recurrent connections whights value of this layer are fixed (Kim and Kim, 2008; Ren et al., 2018).

The output of the hidden layer of Elman network at time t can describe as follow :

$$h_j^{(t)} = g_j \left(b_j + \sum_{i=1}^N x_i^{(t)} w_{ij} + \sum_{i=1}^N h_i^{(t-1)} w_{ij} \right) \quad (I.4)$$

where i, j , = input and hidden layers units index; $x_i^{(t)}$ = input vector for the sequence (t) ; $h_j^{(t)}$ = output hidden layer for the sequence (t) ; $h_j^{(t-1)}$ = output hidden layer for the sequence $(t-1)$; $g_j(.)$ = the transfer function of hidden layer, which provides the output of hidden layer; w_{ij} = the

weight between input and hidden layers; u_{ij} = the weights between hidden layer of sequence (t-1) and hidden layer of sequence (t); b_i = the bias of hidden layer and N = number of sequences.

And the final of output layer at t time as follow:

$$o_k^{(t)} = g_k \left(b_k + \sum_{i=1}^N h_j^{(t)} w_{jk} \right) \quad (I.5)$$

$O_k^{(t)}$ = output of output layer ; $g_k(\cdot)$ = the transfer function of hidden layer; b_k = the bias of output layer; v_{jk} = the weight between output and hidden layers.

I.3.2. 2. Long Short Terme Memory (LSTM)

LSTM is extended version of simple recurrent neural network which effectively enhance the memory. It was first suggested by Hochreiter and Schmidhuber (1997). Unlike the standard RNN, the hidden layers of LSTM model consist of more complicated structures and describe the feedback of information as a chain of repeated simple modules. The fundamental concept of LSTM model is a memory cell that can hold the information regulated over time by specific gate units. LSTM cell (Fig. I.5) can be shown as memory block consisting of input gate, forget gate, and output gate (Cheng et al, 2017).

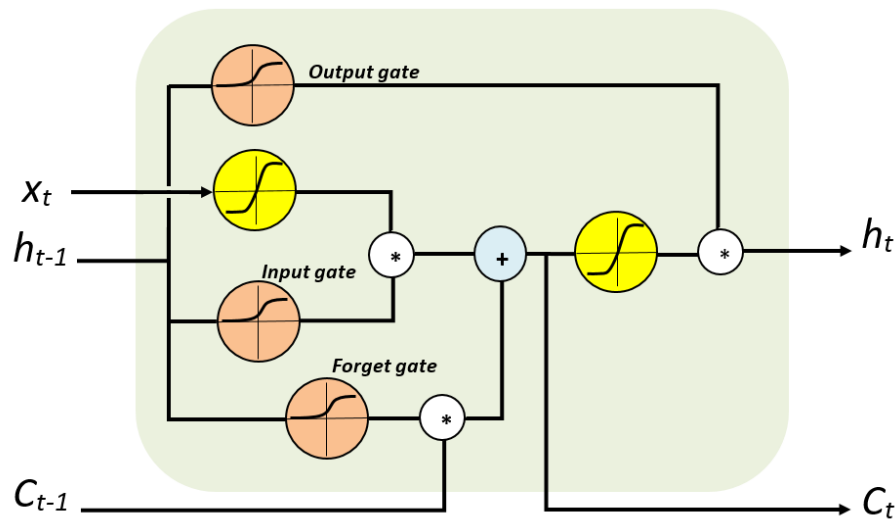


Figure. I.5. Long-short term memory (LSTM) gates processing.

Input gate monitors the activation signal into the memory cell. This segment of network learns the circumstances in which certain information must be maintained or updated. Forget gate aims to reset memory cells by ignoring the past data when specific portions of the cell state are to be replaced with more recent information using an activation function sets zero for values (should be ignored) and 1 for values (should be remembered). Output gate learns and determines which information will be propagated forward to deliver the activations to the final node in the output network

Equations 4-8 explain the detail of all forward pass procedures of the LSTM unit.

$$F_j^{(t)} = \sigma_g \left(b_i^F + \sum_{i=1}^N x_i^{(t)} W_{ij}^F + \sum_{i=1}^N h_i^{(t-1)} U_{ij}^F \right) \quad (\text{I.6})$$

$$I_j^{(t)} = \sigma_g \left(b_i^I + \sum_{i=1}^N x_i^{(t)} W_{ij}^I + \sum_{i=1}^N h_i^{(t-1)} U_{ij}^I \right) \quad (\text{I.7})$$

$$O_j^{(t)} = \sigma_g \left(b_i^O + \sum_{i=1}^N x_i^{(t)} W_{ij}^O + \sum_{i=1}^N h_i^{(t-1)} U_{ij}^O \right) \quad (\text{I.8})$$

$$C_j^{(t)} = F_j^{(t)} * C_j^{(t-1)} + I_j^{(t)} * \sigma_c \left(b_i^C + \sum_{i=1}^N x_i^{(t)} W_{ij}^C + \sum_{i=1}^N h_i^{(t-1)} U_{ij}^C \right) \quad (\text{I.9})$$

$$h_j^{(t)} = O_j^{(t)} * \sigma_h \left(C_j^{(t)} \right) \quad (\text{I.10})$$

where x is the input vector for the time t , (W^F, W^I, W^O, W^C) and (U^F, U^I, U^O, U^C) are the weight matrices of input an hidden layers respectively for different gats, (b^F, b^I, b^O, b^C) are the bias vector for different gats, h is the output vector of the hidden layer, F is the forget gate vector, I is input gate vector, O is the output gate vector, C is the cell state vector, σ_g is sigmoid function, σ_c and σ_h are the hyperbolic tangent functions and $(*)$ denotes the Hadamard product.

I.3.2. 3. Gated Recurrent Unit (GRU) Network

GRU is a state-of-the-art version of recurrent neural network suggested by Cho et al. (2014).

The GRU model is a redesigned concept from the LSTM model based on the gating

mechanism. Compared to the LSTM model, the GRU model reduces the usage of memory cells and merges the forget and input gates into one gate (Gasparin et al., 2019, Yang et al. 2020). The GRU model limits the number of gating signals to two gates (Fig.I.6).

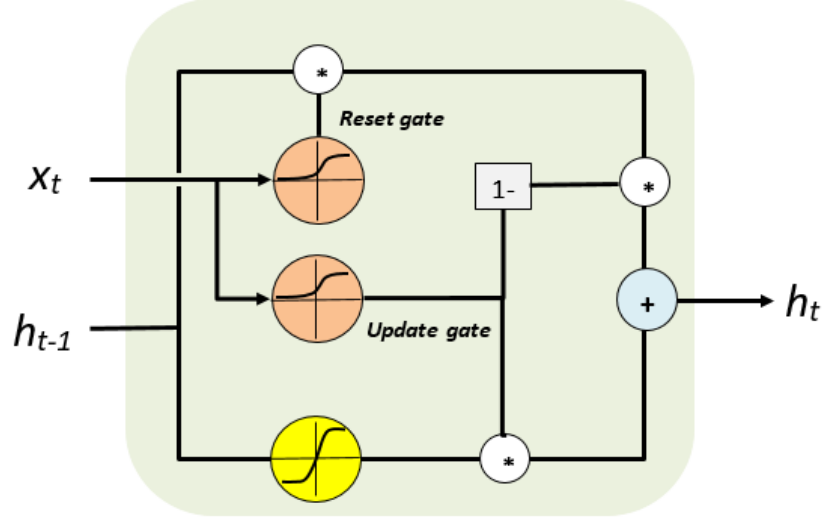


Figure.I.6. Gated recurrent units processing.

Updating the gate determines how often the device changes its configuration or information. The reset gate decides to merge the current configuration information with the historical memory. Also, configuration parameters are used in the GRU model configuration. The calculation equations for the update gate, reset gate, and standard GRU model are as follows:

$$Z_j^{(t)} = \sigma_Z \left(b_i^Z + \sum_{i=1}^N x_i^{(t)} W_{ij}^Z + \sum_{i=1}^N h_i^{(t-1)} U_{ij}^Z \right) \quad (\text{I.11})$$

$$R_j^{(t)} = \sigma_R \left(b_i^R + \sum_{i=1}^N x_i^{(t)} W_{ij}^R + \sum_{i=1}^N h_i^{(t-1)} U_{ij}^R \right) \quad (\text{I.12})$$

$$H_j^{(t)} = \sigma_H \left(b_i^H + \sum_{i=1}^N x_i^{(t)} W_{ij}^H + \sum_{i=1}^N h_i^{(t-1)} [U_{ij}^H * R_j^{(t)}] \right) \quad (\text{I.13})$$

$$h_j^{(t)} = \left(1 - Z_j^{(t)} \right) h_j^{(t-1)} + Z_j^{(t)} H_j^{(t)} \quad (\text{I.14})$$

where x = the input vector at time t ; (W^Z, W^R, W^H) and (U^Z, U^R, U^H) = the weight matrices of input and hidden layers for different gates. σ_H = the hyperbolic tangent function; σ_Z and σ_R = the sigmoid functions; and $*$ = the Hadamard product.

I.3.3. Adaptive neuro-fuzzy inference system (ANFIS)

Neuro-fuzzy systems based on the integration of neural networks along with fuzzy inference systems (FIS). The basic structure of the type of fuzzy inference system can be seen as a model that maps input characteristics to input membership functions (MFs) (Moosaviet al, 2013). For building a FIS, the following processes are required: (1) fuzzification: the FIS employs linguistic rules of the type (If-Then) which translate knowledge about the dynamic of a system (Komori, 1992), (2) fuzzy database: defining the MFs, and (3) defuzzification: an inference system that combines the fuzzy rules and produces the system results. The learning procedure of neuro-fuzzy systems is carried out on account of the local information and introduces only local changes to the original fuzzy system. Among various neuro-FIS, Takagi–Sugeno (TS) systems (Takagi & Sugeno, 1985) have been applied successfully for data-driven based fuzzy modeling. The TS approach has either a scalar or a function of input variables as the consequent part of the If–Then inference rule. ANFIS is a multi-layer adaptive network-based (TS) fuzzy model proposed by Jang (1993). It has five functional blocks, which are generated using five layers of neurons. To explain the computations involved, we consider a simple fuzzy inference system with two inputs x_1 and x_2 , one output y , and two MFs in each input. A typical rule set for a first-order Sugeno-fuzzy model (Lohani et al, 2006; Nasr & Bruen, 2008; Nayak et al, 2004; Talei et al, 2010; Yurdusev & Firat, 2009) that includes four fuzzy If-Then rules can be expressed as:

R₁: If x_1 is $\mu_1^{(1)}$ (and/ or) x_2 is $\mu_1^{(2)}$ Then $y_1 = f_1(x_1, x_2)$

R₂: If x_1 is $\mu_2^{(1)}$ (and/ or) x_2 is $\mu_2^{(2)}$ Then $y_2 = f_2(x_1, x_2)$

R₃: If x_1 is $\mu_3^{(1)}$ (and/ or) x_2 is $\mu_3^{(2)}$ Then $y_3 = f_3(x_1, x_2)$

R₄: If x_1 is $\mu_4^{(1)}$ (and/ or) x_2 is $\mu_4^{(2)}$ Then $y_4 = f_4(x_1, x_2)$

Figure (I.7) represents the principal operation of adaptive neural-fuzzy inference system

(ANFIS) that is based on the following layers:

Layer 1: every node in this layer is an adaptive node with a node function named MF.

$$\mu_i^{(j)}(x, \delta): x \rightarrow [0, 1] \quad (\text{I.15})$$

$$i = 1 \dots 4, j = 1 \dots 2.$$

Layer 2: nodes in this layer are labeled F, whose output represents a firing strength of a rule.

The node generate the output as follow:

- For the (And) rules we have two choices:

$$w_i = F(\mu_i^{(1)}, \mu_i^{(2)}) = \begin{cases} \min(\mu_i^{(1)}, \mu_i^{(2)}) \\ \text{or} \\ \prod(\mu_i^{(1)}, \mu_i^{(2)}) \end{cases} \quad i = 1 \dots 4 \quad (\text{I.16})$$

- For the (Or) rules we have also two choices:

$$w_i = F(\mu_i^{(1)}, \mu_i^{(2)}) = \begin{cases} \max(\mu_i^{(1)}, \mu_i^{(2)}) \\ \text{or} \\ \text{probOR}(\mu_i^{(1)}, \mu_i^{(2)}) \end{cases} \quad i = 1 \dots 4 \quad (\text{I.17})$$

Layer 3: the i th node of this layer, labeled as N, normalized with respect to the other input,

where the output node i is equal to the input i divided by the sum of the inputs.

$$\bar{w}_i = w_i / \sum w \quad i = 1 \dots 4 \quad (\text{I.18})$$

Layer 4: the output of the node i is a linear function of the output and the signals input of the controller.

$$\bar{w}_i y_i = \bar{w}_i (p_i x_1 + q_i x_2 + s_i) \quad i = 1 \dots 4 \quad (\text{I.19})$$

Layer 5: the summation of the signals input.

$$\tilde{y} = \sum \bar{w}y = \frac{\sum wy}{\sum w} \quad (\text{I.20})$$

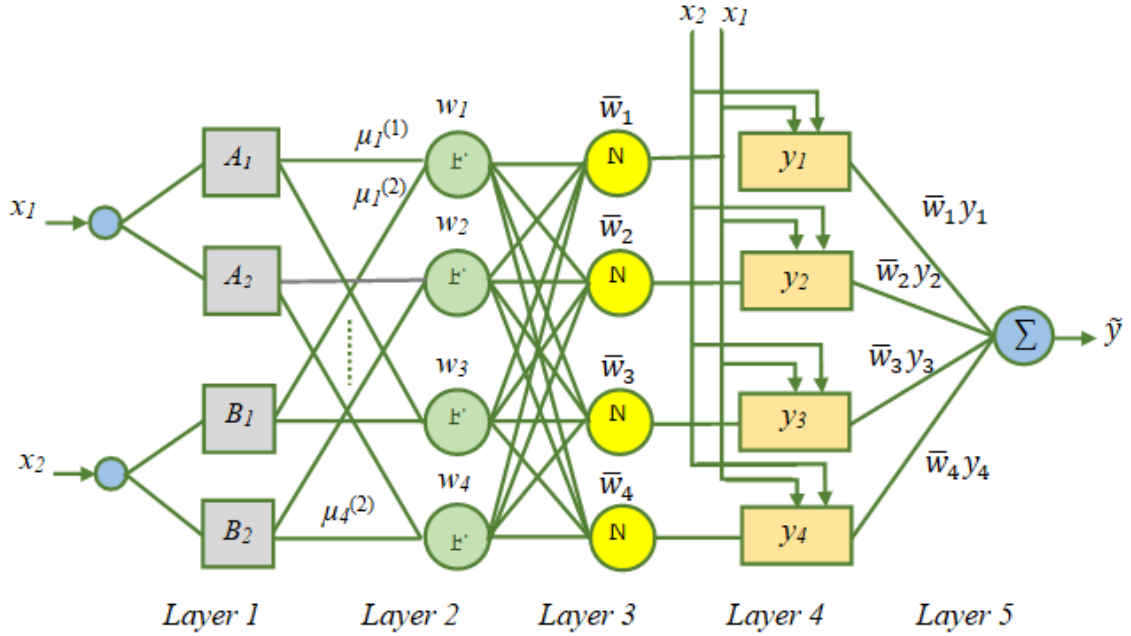


Figure. I.7. Operation principle of the ANFIS network based on the fuzzy inference system, Takagi–Sugeno (TS) type.

I.3.4. Wavelet-based neural networks (WNN)

I.3.4. 1. Wavelet transform (WT)

A wavelet is a “small wave” function, usually named “mother wavelet.” The mother function can be used to generate a whole family of wavelets by translating and scaling the mother wavelet. A wavelet transformation is a signal processing tool, like Fourier transformation, with the ability of analyzing both stationary as well as non-stationary data (Pralhada & Deka, 2011). The basic objective of the WT is to achieve a complete time scale representation of localized and transient phenomena occurring at different time scales (Labat et al, 2000). The power-of-two logarithmic scaling of the dilations and translations is known as a dyadic grid arrangement, and is the simplest and most efficient case for practical purposes (Mallat, 1989).

The signal time series is decomposed into one comprising low frequencies and its trend (the approximation), and one comprising the high frequencies and fast events (the detail). The detail signals can capture small features of interpretative value in the data; the approximation represents the background information of data (Fig. I.8).

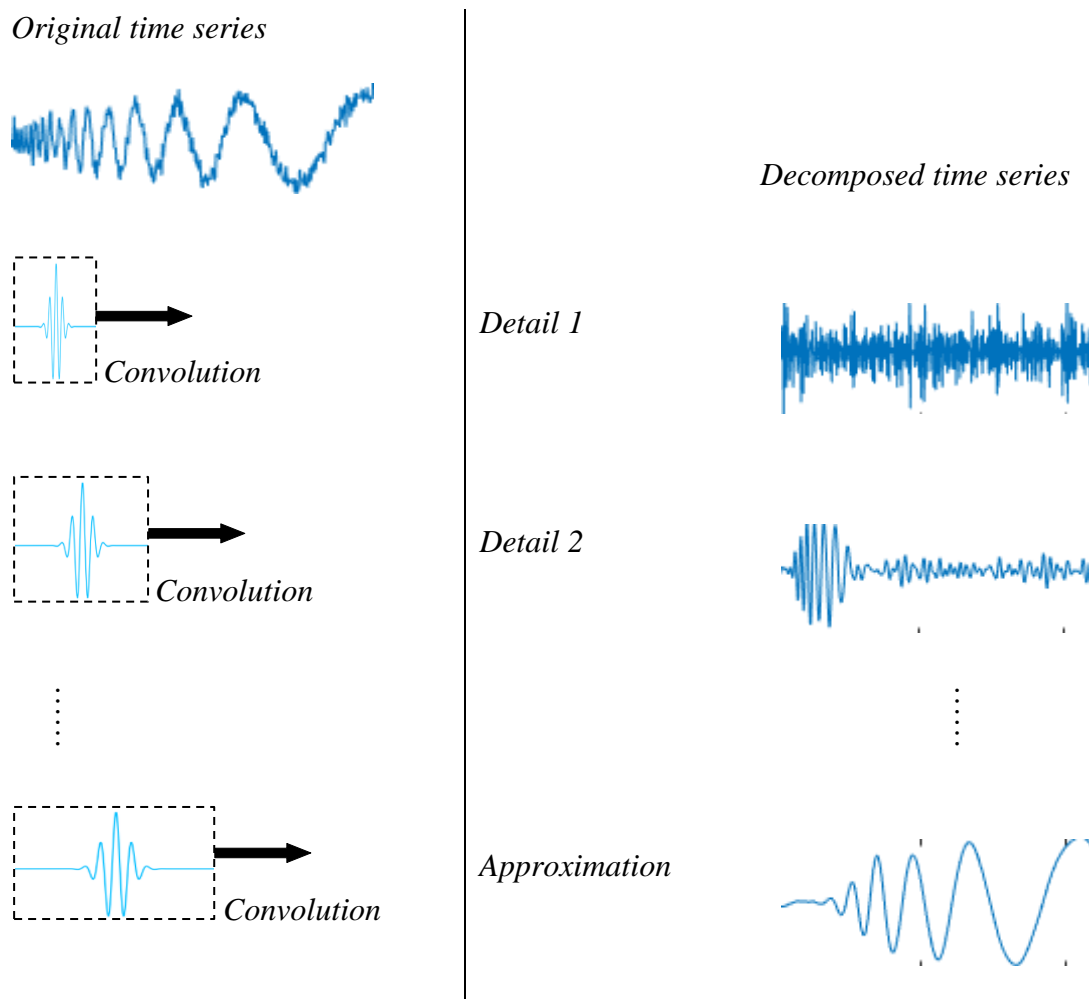


Figure.I.8. Wavelet transform mechanism.

I.3.4. 2. Discrete wavelet transform (DWT)

WT is a mathematical function which is capable of decomposing the continuous or discrete time signal into the higher- and low-frequency components of a signal (Mallat 1989). There are two main categories of wavelet transforms [e.g., continuous wavelet transform (CWT) and discrete wavelet transform (DWT)], respectively. CWT method treats with continuous functions and can be applied for discrete functions or time series (Seo et al. 2015, 2018; Nourani et al. 2009; Mallat 1989). In brief, CWT method is time-consuming and requires large resources, while DWT method can be applied than CWT method easily (Nourani et al. 2009; Mallat 1989). DWT method consists of two decompositions (e.g., lowpass and high-pass) and reconstructions (e.g., low-pass and high-pass), respectively (Seo et al. 2015, 2018; Mallat 1989). For practical application of DWT method, two filters (e.g., low-pass and high-pass) are utilized rather than two wavelets (e.g., father and mother) (Seo et al. 2015). The low-pass filter allows for the analysis of low-frequency components, while the high-pass filter allows for the analysis of high-frequency components (Seo et al. 2015). The multi-resolution approach utilizing DWT method is a process to draw ‘approximations (show a conventional trend of original signal)’ and ‘details (indicate the high-frequency components)’ for a given signal (Seo et al. 2015, 2018; Mallat 1989). The feature report for DWT method can be collected from Nason (2008). So we found the Discrete wavelet transform (DWT) by using the follow equations :

$$\begin{cases} a = 2^j \\ b = n2^j \\ \Psi_{m,n} \left(\frac{t-b}{b} \right) = a_0^{\frac{-m}{2}} \left(\frac{t-nb_0a_0^m}{a_0^m} \right) \end{cases} \quad (\text{I.21})$$

$$W(i, m) = 2^{\frac{-i}{2}} \sum_{n \in \mathbb{Z}} x(n) \Psi \left(\frac{n}{2^i} - m \right) \quad (\text{I.22})$$

where j and n are the entiers numbers, a is the scall factor , b is the translation parametr and $\Psi(t)$ is the mother wavelet.

Some of the most important mother wavelets using for DWT are Haar, Daubechies (db), Symlets (sym), Coiflets (Coif)...ect.

A WNN model is a combination of Wavelet transform WT and artificial neural networks models ANN, where the original time series for each input neuron of the ANN model is decomposed to some multi-frequency time series using the WT algorithm. The decomposed details (D) and approximation (A) are formatted as new inputs to the ANN model architecture, as shown in Fig(I.9). In the WNN, the number of decomposition levels (details and approximation) of each input is calculated according to full data length (Nourani et al., 2009).

$$L = \text{int}[\log(N)] \quad (\text{I.23})$$

where L defines the decomposition level, N denotes the number of time series data, and $\text{int}[\cdot]$ depicts the integer-part function.

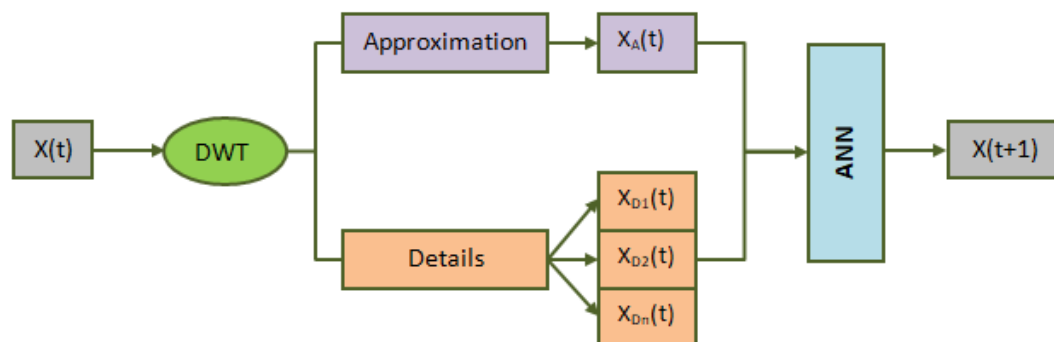


Figure.I.9. The schematic diagram of WNN model.

I.4. Conclusion

In this chapter, the background theory of artificial neural networks was briefly reviewed.

Chapter. II

Applied Neuro-Wavelet and Neuro-Fuzzy systems on the catchment of Ain Safra.

II.1. Introduction

Sustainable development of human activities is based especially on an integrated management of water resources. Hence, an efficient and sustainable management of water resources cannot be limited to mere guarantee of sufficient quantity and quality of water to meet the needs of humans (drinking, industry, irrigation, etc.), for it has to take into account the occurrence of extreme events, such as drought flow and flooding. Like most countries on the southern shore of the Mediterranean, Algeria, whose climate is essentially semi-arid to arid in the major parts of its territory, is facing issues in development and management of its water resources. The transformation of rainfall into runoff is the result of a number of complex mechanisms that are to take place simultaneously at different spatial and temporal scales (Fortin et al,1997). However, developing a rainfall–runoff model becomes a necessity in that, it is designed to take into account the recorded data of rainfall which may enable the model to produce a runoff as close as possible to the recorded data; in other words, we can reproduce (or predict) the response in terms of runoff of the basin based on the records of rainfall. During the last 20 years, a large number of approaches were carried out for the purpose of modeling the process of the transformation of rainfall into runoff.

However, the complexity of the hydrologic regimes requires the use of specific tools of non-linear dynamic systems (Sivakumar et al, 2001). In this respect, we propose in this work, in order to model this process, to use a wavelet based feed forward neural network (WFFNN) and adaptive neuro-fuzzy inference system (ANFIS). The aim behind system modeling incorporating neural networks and (wavelet/fuzzy inference systems FIS) lies in the fact that their characteristics are complementary. Using the discrete wavelet transform, a series of rainfall and streamflow are decomposed into a succession of approximation and details and used as inputs in a model. The FIS make use of linguistic rules that translate knowledge of the dynamic of a system. The performance of the adaptive Wavelet Neural Network (WNN) and neuro-fuzzy inference system (ANFIS) has been proved in several fields of engineering and

science. The most recent studies using WNN and neuro-fuzzy systems to model the rainfall–runoff relationship, for example, are those of the authors in references (Aqil et al, 2007; Chettih et al, 2012; Dastorani et al, 2010; Krishna et al, 2011; Nasr et al, 2008; Talei et al, 2010; Tiwari et al, 2010;). This chapter includes partial contributions from the paper (Zakhrouf et al, 2015).

II.2. Materials and Methods

The models used in this chapter are : Feed forward neural network (FFNN) (see paragraphe Feed forward neural network in chapter I), Adaptive Network-based Fuzzy Inference System (ANFIS) (see paragraphe Adaptive Network-based Fuzzy Inference System in chapter I) , Wavelet Feed forward neural network (see paragraphe Wavelet neural network in chapter I)

II.3. Study area and data used

The Ain Sefra watershed is forms the upstream part of the large catchment area of the Oued Namous of the Sahara Basin, it is located in the south-west of Algeria between longitudes ($1^{\circ} 0' 0''$ and $0^{\circ}03'00''$ W) and latitudes ($32^{\circ}30'22''$ and $33^{\circ}00'00''$ N). The Ain Sefra watershed covers an area of 1957 km² for a perimeter of 236 km, with a compactness index of 1.49, which characterizes a moderately elongated basin (Fig. II.1). The Ain Sefra region is characterized by a semi-arid climate, with dry and hot summers where rainfall is almost absent and evaporation particularly strong (Derdour et al, 2016).

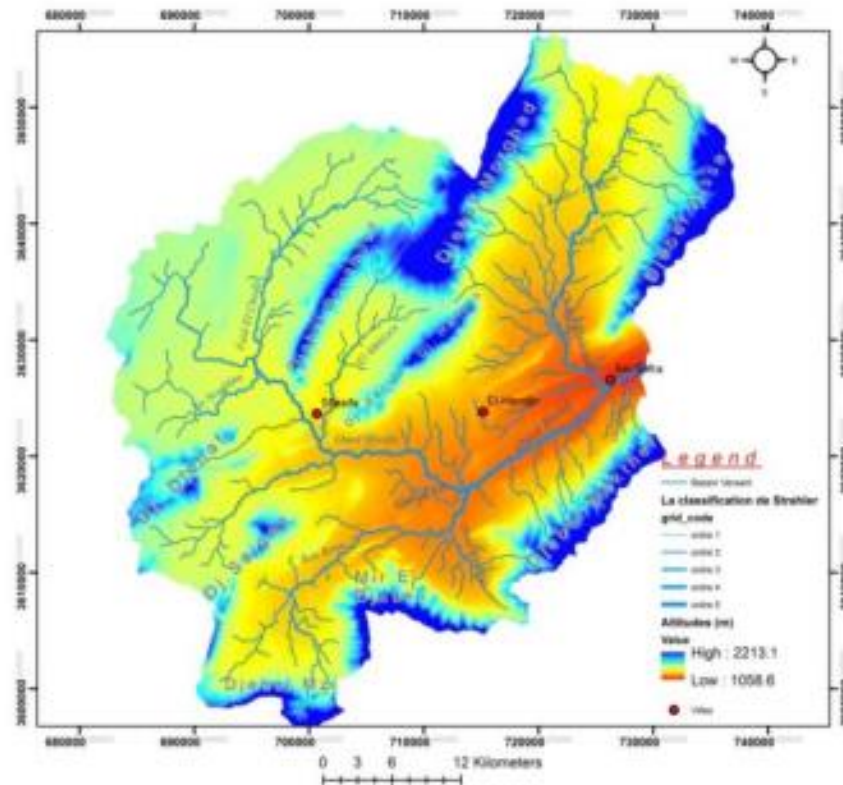


Figure. II.1. Ain Safra watershed (Derdour et al).

The database contains values of rainfall and daily streamflow of Aï'n Hadjadj watershed (Fig. 6). The Aï'n Hadjadj watershed is located in the Saharan Atlas. The hydrometric station of Aï'n Hadjadj coded (r 0,345) by the National Agency of Water Resources, is the feeding source for the other sources such as: Aï'n Melalek, Ain Esomam, Aï'n Tessala, and Aï'n Skhouna. The hydrometric data represent a chronicle of 25 years from 1 September 1973 until 31 August 1997. The rainfall station used for the study of the rainfall–runoff relationship in the Aï'n Hadjadj basin is in the city of Aï'n Sefra. The database was divided into two sets: (1) a set for the training phase of the model corresponding to 60% of the data; (2) the other set for the testing phase of the model corresponding to the remaining 40% (Fig. II.2).

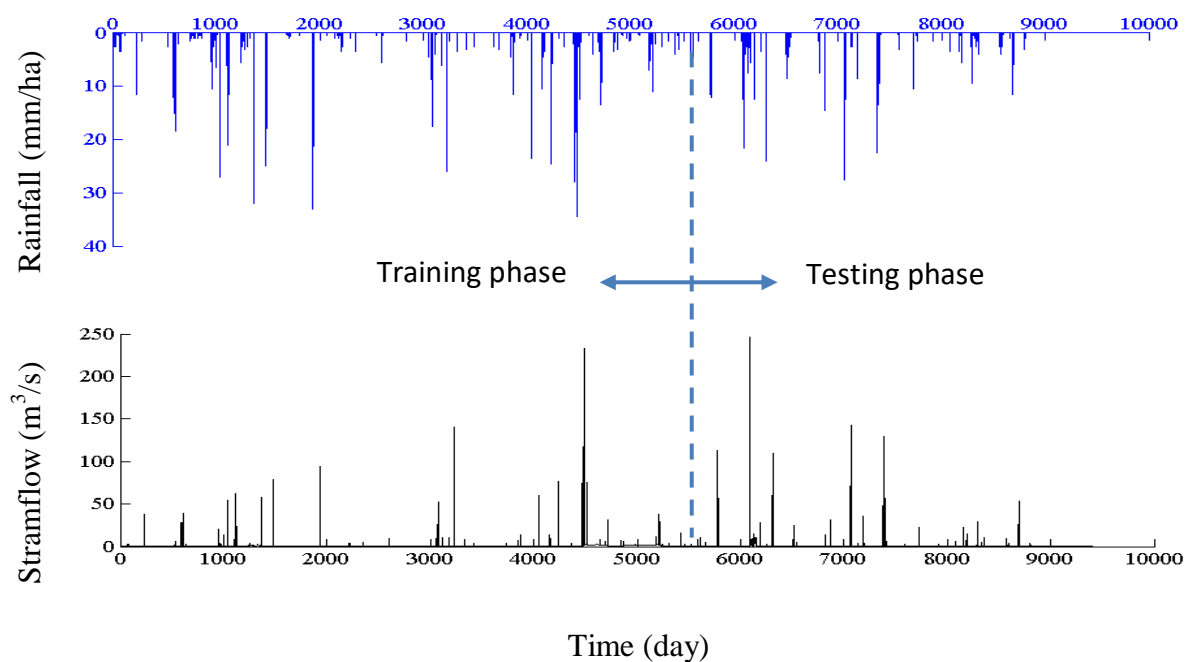


Figure. II.2. Average daily streamflow and cumulative rainfall measured at stations: Ain Hadjadj - Ain safra, (period 1973/1999).

II.4. Input selection

Based on the work done in the field of hydrology, the input parameters of the models are the observed rainfall and runoff in previous instants (t , $t - 1$, $t - 2$, ..., $t - n$) (Nayak et al, 2004, ErolKeskin 2010).

Consequently, the output of the models represents the predicted value of the streamflow for the next day ($t + 1$), i.e.:

$$Q_{t+1} = f(P_t, P_{t-1}, \dots, P_{t-n}, Q_t, Q_{t-1}, \dots, Q_{t-n}) \quad (\text{II.1})$$

Using the correlogram and variance spectral density, the time series of rainfall and discharge of a hydrological system are analyzed in a descriptive way (Mangin, 1984). The results obtained by the correlogram and the density spectrum in short term for the watershed are shown in Figs. (II.3 and II.4), where they showed the absence of memory effects which modulated the input rainfall for short term (Figs. II.3.a and II.3.a), they should highlight a

rapid decrease of correlogram, the values oscillate around zero after the tenth day, the density spectrum of variance shows a fairly regular decrease which could be explained by the dependence of successive variables (Fig.II.3.a and II.3.b). In this case, Ar'n Hadjadj system is considered without memory, only the rainfall and the runoff measures of the same day were exploited, i.e.:

$$Q_{t+1} = f(P_t, Q_t) \tag{II.2}$$

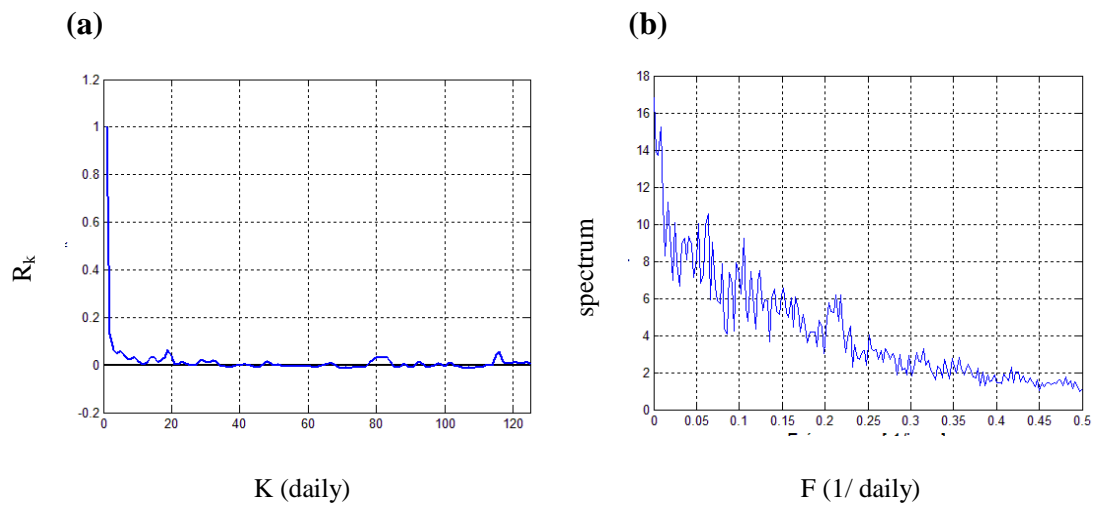


Figure.II.3. (a) Simple Correlogram of rainfall, (b) density variance spectrum of rainfall

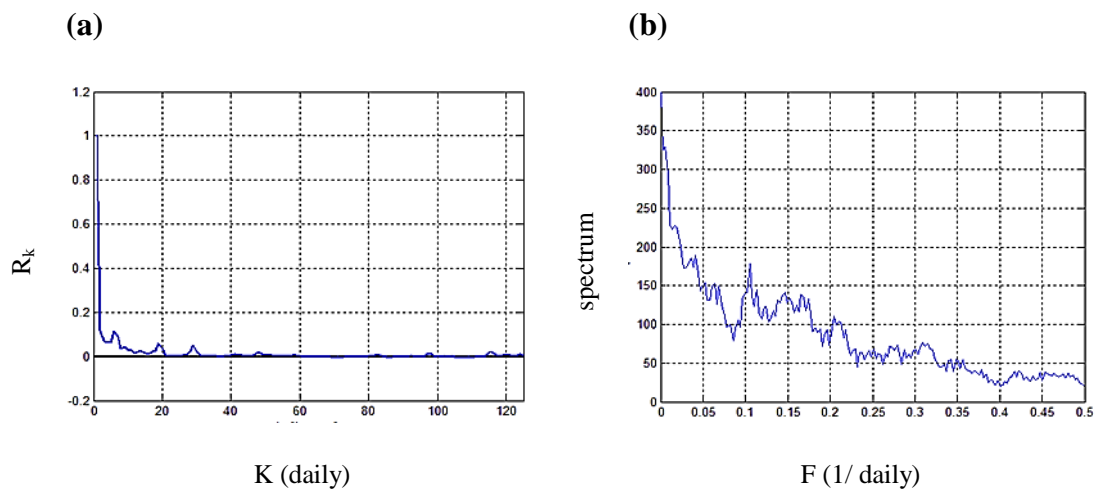


Figure.II.4. (a) Simple Correlogram of stramflow, (b) density variance spectrum of stramflow.

II.5. Implementation of ANN and WNN

A neural network with two layers having a sigmoidal activation function in the first layer (hidden layer) and a linear function in the output layer allows to approximate any function of interest with arbitrary precision, provided that there are enough neurons in the hidden layer (Aqil et al, 2007, Wang et al, 2004). The optimal number of neurons in the hidden layer has been identified through trying and error method varying the number of the hidden neurons. In this case, we start with architecture of 1 neuron in the hidden layer, and constantly increasing this number up to 16 neurons. Then, we take the architecture that gives the minimum error on the test phase. In our study, several publications show that the Levenberg–Marquardt Backpropagation (LMBP) algorithm gives the most efficiency (Aqil et al, 2007, Alpaslan et al, 2009, Wu et al, 2010).

With the LMBP optimization algorithm, an error function measuring the difference between the observed (y_k) and expected (d_k) outputs can be minimized as follow:

$$E(w, x, d_k) = \frac{1}{2} \sum_1^k (d_k - y_k)^2 \quad (\text{II.3})$$

This algorithm involves two steps: (1) a "forward move" during which network outputs are estimated from inputs and (2) a "backward move" during which partial derivatives of a certain cost function E with respect to parameters are replicated. Finally, the weights are modified according to the following equations:

$$w_{ij}(n) = -(J_n^T J_n + \mu I)^{-1} J_n^T E(n) + w_{ij}(n - 1) \quad (\text{II.4})$$

Where J = the Jacobian matrix of E , I = the identity matrix, and μ = learning parameter, w = the weights and n = the iterations.

The effect of the number changing in hidden neurons on the quality of results is shown in Fig (II.5), it can be deduced that the optimal number of 11 neurons gives us the best results to model streamflow.

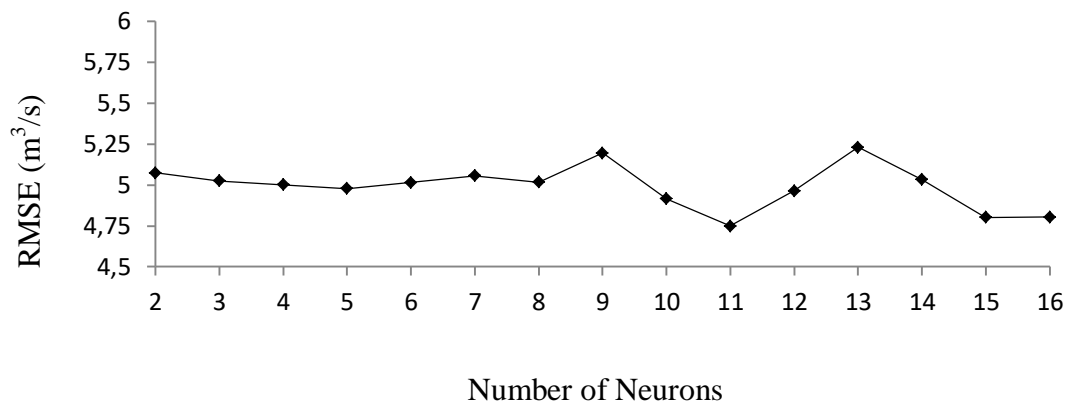


Figure.II.5. The optimal number of neurons for FFNN model in the testing phase.

For WFFNN, the wavelet decomposition can be iterated, with successive approximations being decomposed in turn, so that the signal is broken down into many lower resolution components, tested using different scales from 1 to 10 with different sliding window amplitudes. In this context, dealing with a very irregular signal shape, an irregular wavelet, the Daubechies wavelet of order 5 (DB5), has been used at level 10. The effect of the number changing in hidden neurons on the quality of results is shown in Fig(II.6), according to the examination of Fig(II.6), it can be deduced that the optimal number of 8 neurons gives us the best results to model streamflow.

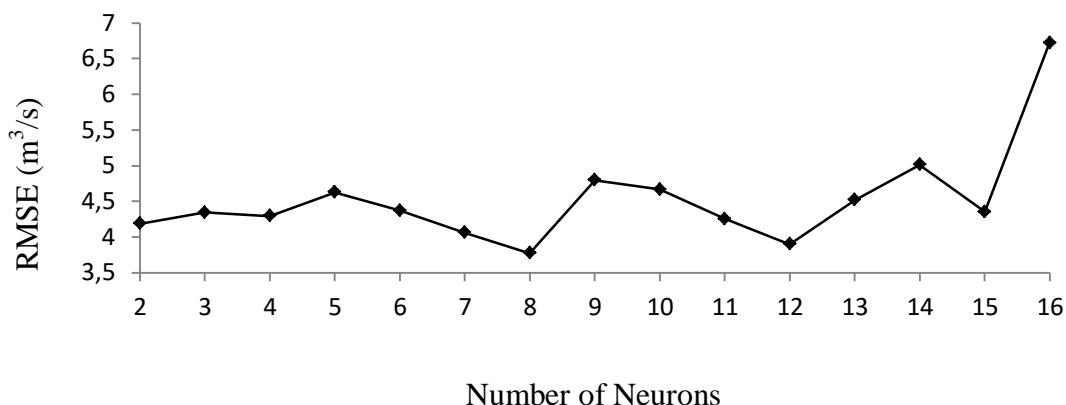


Figure.II.6. The optimal number of neurons for WFFNN model in the testing phase.

II.6. Implementation of ANFIS

In the ANFIS model, each rule contains some parameters of membership functions (MFs) and each variable may have some values (in terms of rules). For example, if each variable has two rules and each rule contains three parameters, then there are $6n$ parameters (n variables \times 2 rules \times 3 parameters) for the determination in layer 1 (see chapter I). The ANFIS model calibrates these MFs in relation to calibration data. These rules produce $2n$ nodes in layer 3. In this part, the number of MFs varying from 2 to 7 was examined. The hybrid-learning approach in the neuro-fuzzy model can be employed for a search of the optimal parameters of the ANFIS. During the learning process, the premise parameters of membership functions δ , and the consequent parameters θ are tuned until the desired response of the FIS is achieved.

The learning rule specifies how the premise parameters and consequent parameters should be updated to minimize a prescribed error measure.

$$E(\theta, \delta) = \frac{1}{k} \sum_1^k (d_k - y_k)^2 \quad (\text{II.5})$$

Where y_k = observed outputs and d_k = expected outputs

The hybrid learning algorithms of ANFIS consist of the following two parts.

In the forward pass of the hybrid learning algorithm, the least square method can be used to determine the optimally values of the consequent parameters; the overall output can be expressed as a linear combination of the consequent parameters as follow:

$$\theta^* = (A^T A)^{-1} A^T Y \quad (\text{II.6})$$

Where A^T is the transpose of A and $(A^T A)^{-1} A^T$ is the pseudo inverse of A if $A^T A$ is nonsingular.

The nonlinear or premise parameters in the layer 2 remain fixed in this pass. In the backward pass, the error rates propagate backward from the output end towards the input end, and the premise parameters are updated by the gradient descent. The update formula for simple steepest descent for the generic parameter δ is:

$$\delta_{k+1} = \delta_k - \alpha \frac{\partial E}{\partial \delta} \quad (\text{II.7})$$

Where, α = learning rate and $\partial E / \partial \delta$ = partial derivative of the function E .

Gaussian membership functions are used for each fuzzy rule in the ANFIS system. This choice of functions is based on research work done by (Gautan and Holz, 2001) and (Lohani et al, 2006).

The effect of number change (MFs) on the quality of the results is shown in Fig(II.7), the ANFIS model having 4 MFs, have estimated minimum error.

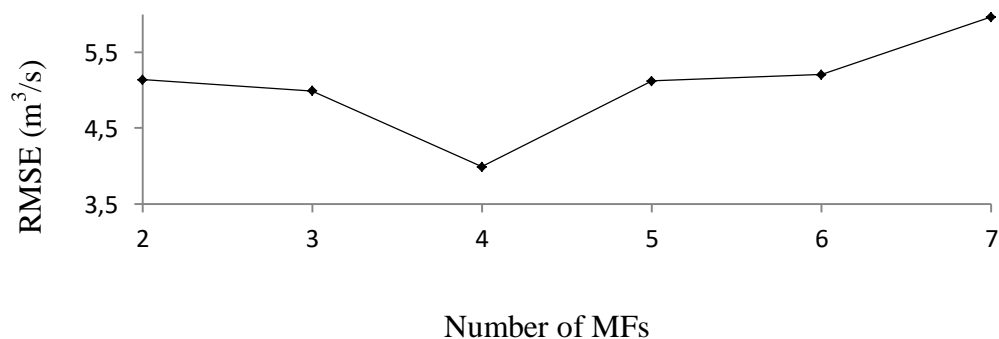


Figure.II.7. The optimal number of MFs for ANFIS model in the testing

II.7. Performance criteria

The statistical parameters used in this work are: the root mean square error (RMSE), the Nash-Sutcliffe efficiency coefficient (NSE) (Abrahart et al, 2004) and the determination coefficient (R^2). These parameters are given by the following relationships:

$$RMSE = \sqrt{\sum_{i=1}^N (Qt_i - \hat{Q}t_i)^2 / N} \quad (II.8)$$

$$NSE = \left(1 - \frac{\sum_{i=1}^N (Qt_i - \hat{Q}t_i)^2}{\sum_{i=1}^N (Qt_i - \bar{Q}t)^2}\right) \cdot 100 \quad (II.9)$$

$$R^2 = \left(\frac{\sum_{i=1}^N (Qt_i - \bar{Q}t)(\hat{Q}t_i - \bar{Q}t)}{\sqrt{\sum_{i=1}^N (Qt_i - \bar{Q}t)^2 \sum_{i=1}^N (\hat{Q}t_i - \bar{Q}t)^2}}\right)^2 \quad (II.10)$$

Where Qt_i is the measured flow rate value, $\hat{Q}t_i$ is the flow rate calculated by the model, $\bar{Q}t$ is the average flow measured, $\bar{\hat{Q}}t$ is the average flow simulated and N is the number of data.

II.8. Results and discussion

The performances of WFFNN, ANFIS, and FFNN in terms of the performance indices are presented in Table (II.1). To have a true evaluation of the potential of WFFNN compared to ANFIS and FFNN models during training phase. Table (II.1) suggests that though the performance of both the WFFNN and the ANFIS models are similar during testing phase, the WFFNN model show a slight improvement over the ANFIS model. It is evident from Table (II.1) that the WFFNN model outperforms the FFNN model in terms of all performance indices. Figs (II.8.a, II.9.a, and II.10.a) show the observed and simulated hydrographs for WFFNN, ANFIS, and FFNN models during the testing phase, It was found that values simulated from WFFNN and ANFIS models correctly matched with the observed values,

whereas, FFNN model underestimated the observed values. The distribution of error along with the magnitude of streamflow, computed by WFFNN, ANFIS, and FFNN models during the testing phase, has been presented in Figs (II.8.b, II.9.b, and II.10.b). From Figs. (II.8.b, II.9.b, and II.10.b), it was observed that the estimation of streamflow was good using WFFNN and ANFIS, because the error was minimum compared to the FFNN model.

Figs (II.8.c, II.9.c, and II.10.c) shows the scatter plot between the observed and modeled streamflow by WFFNN, ANFIS, and FFNN models during the testing phase. It was observed that the streamflow forecasted by WFFNN model was close to the 45° line. From this analysis, it was worth to mention that the performance of WFFNN was much better than ANFIS and FFNN.

	Training phase			Testing phase		
	RMSE m ³ /s	NSE (%)	R ²	RMSE m ³ /s	NSE (%)	R ²
WFFNN	0.461	99.27	0.9920	3.772	77.89	0.802
ANFIS	1.249	94.66	0.9467	3.992	75.25	0.794
FFNN	3.405	60.37	0.6448	4.750	64.96	0.630

Table. II.1. Results obtained by the models: WFFNN, ANFIS and FFNN.

Figs (II.11, II.12 and II.13) show the observed and simulated peak streamflow hydrographs, and relative peak error in each year for WFFNN, ANFIS, and FFNN models. It was observed that WFFNN and ANFIS models estimated the peak value of river flow to a reasonable accuracy (peak flow during the study was 250 m³ /s of year 1989) (Figs. 15 and 16), but from Fig. 17, it was observed that FFNN model is not well-trained, and simulated peak values consistently underestimated the observed peak values.

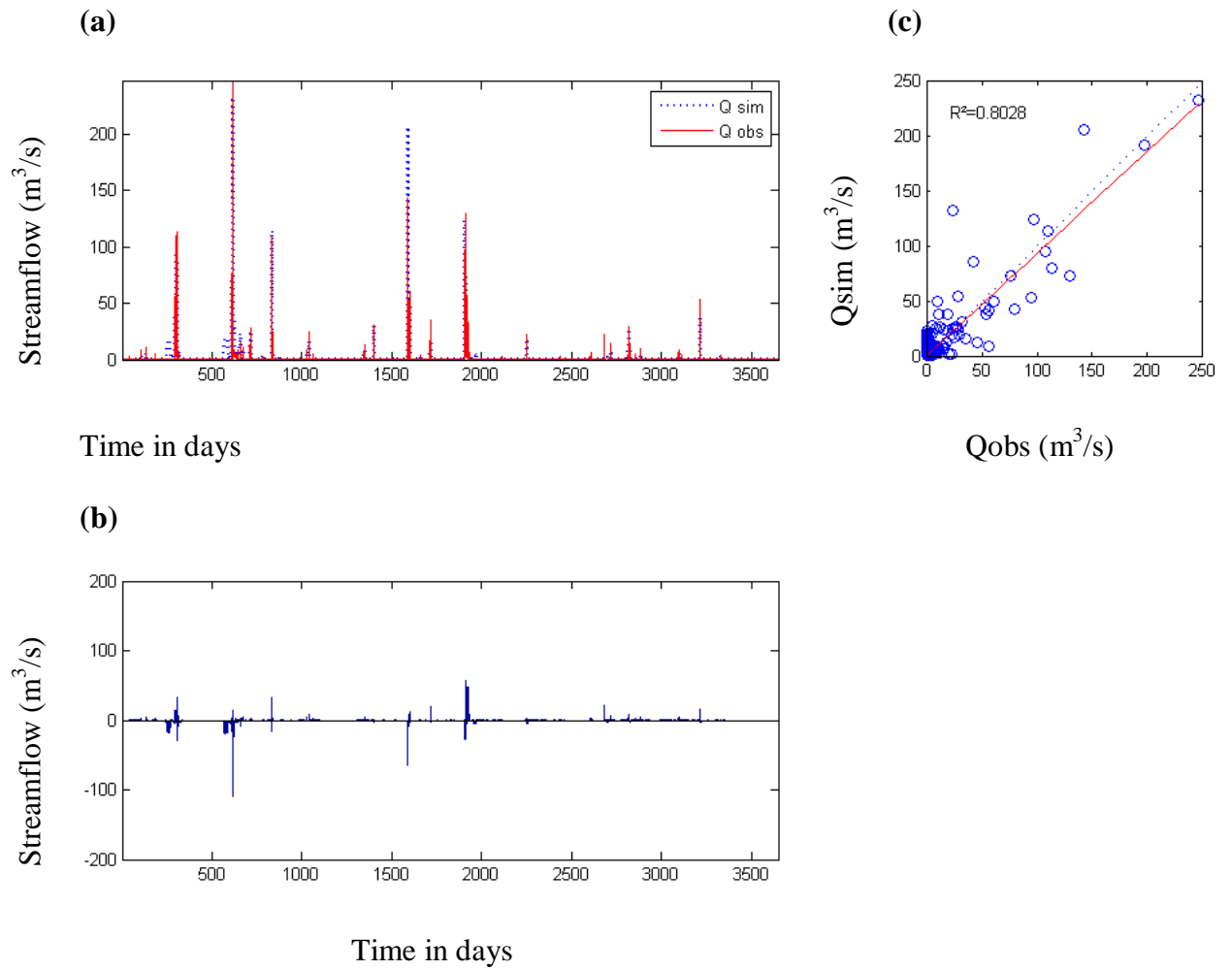


Figure.II.8. Plot of (a). Observed and simulated hydrographs, (b). Error plots along the magnitude of streamflow, (c). Scatter of observed and simulated streamflow for WWFNN model during testing phase.

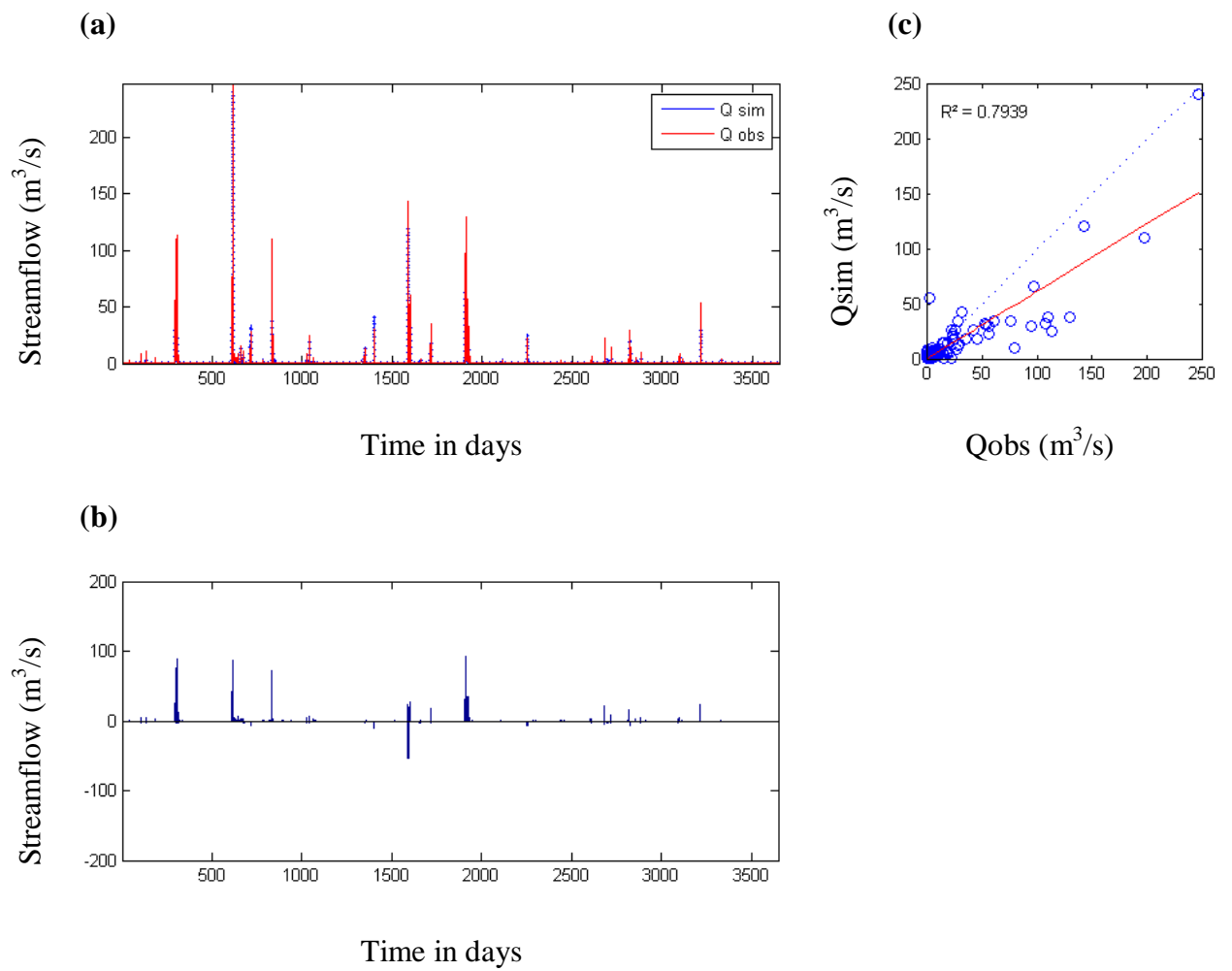


Figure.II.9. Plot of (a). Observed and simulated hydrographs, (b). Error plots along the magnitude of river streamflow, (c). Scatter of observed and simulated streamflow for ANFIS model during testing phase.

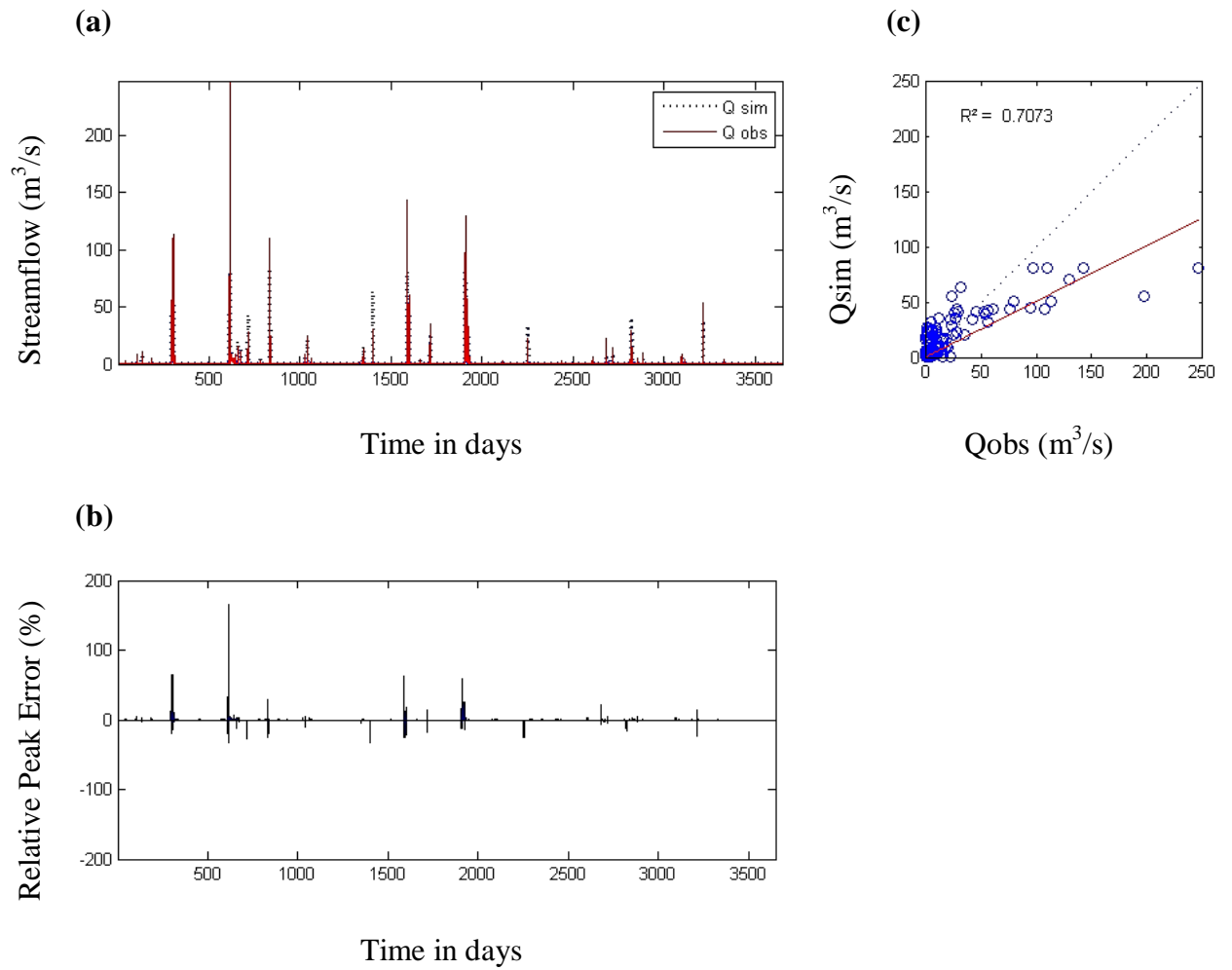


Figure.II.10. Plot of (a). Observed and simulated hydrographs, (b). Error plots along the magnitude of streamflow, (c). Scatter of observed and simulated streamflow for FFNN model during testing phase.

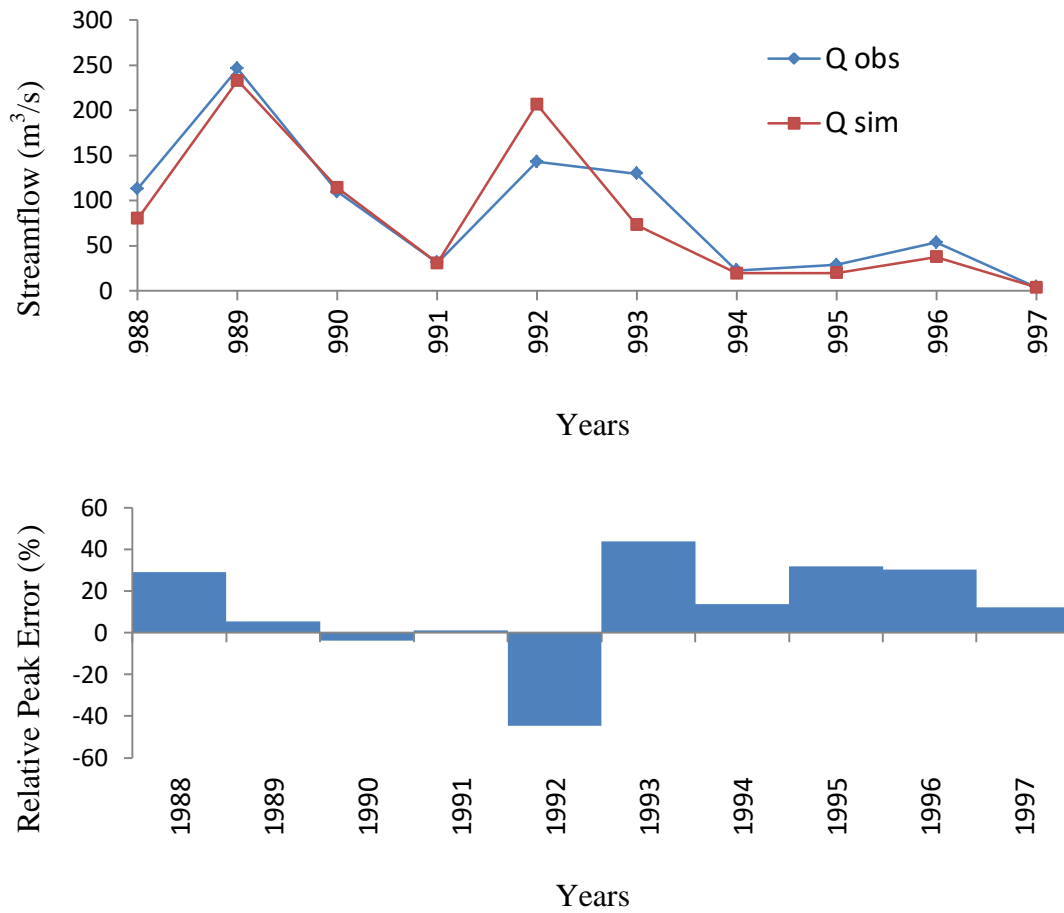


Figure.II.11. Peak flow estimate and relative peak error for WFFNN model during testing phase.

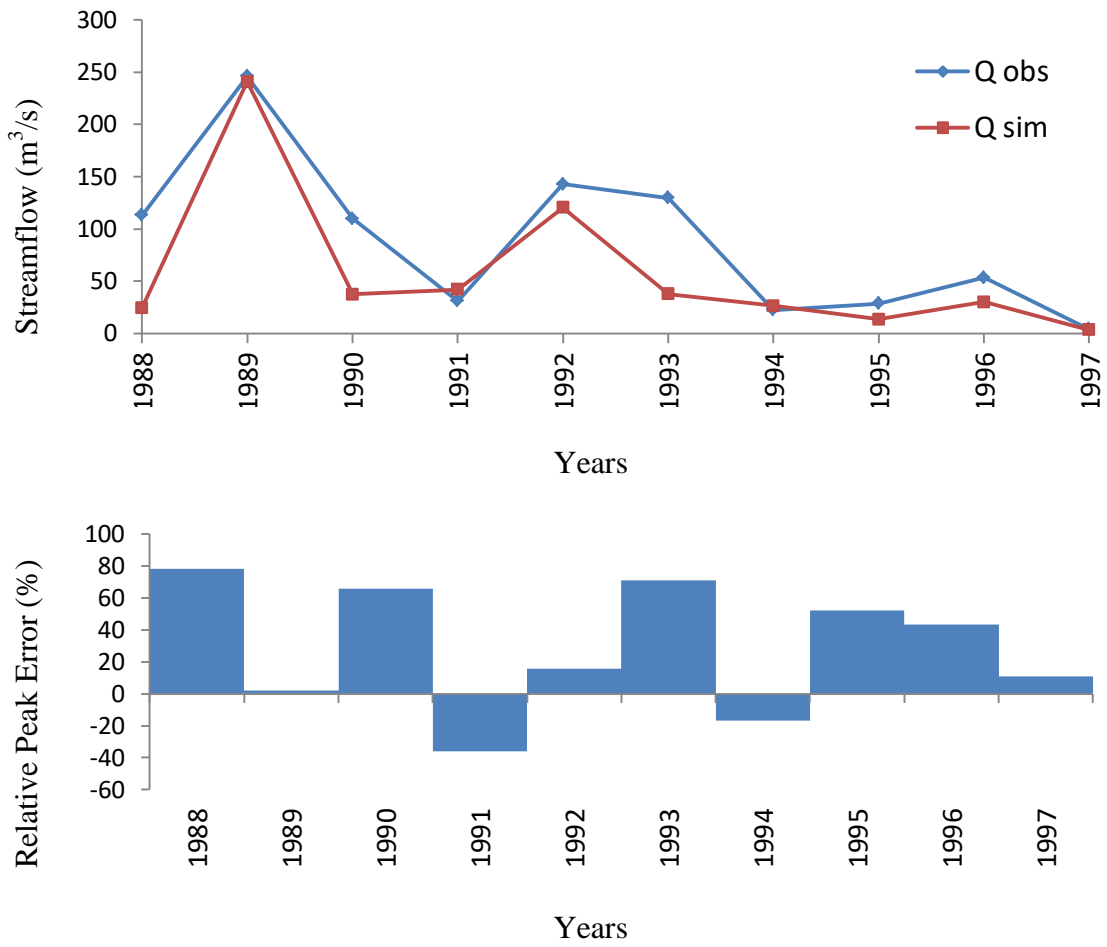


Figure.II.12. Peak flow estimate and relative peak error for ANFIS model during testing phase.

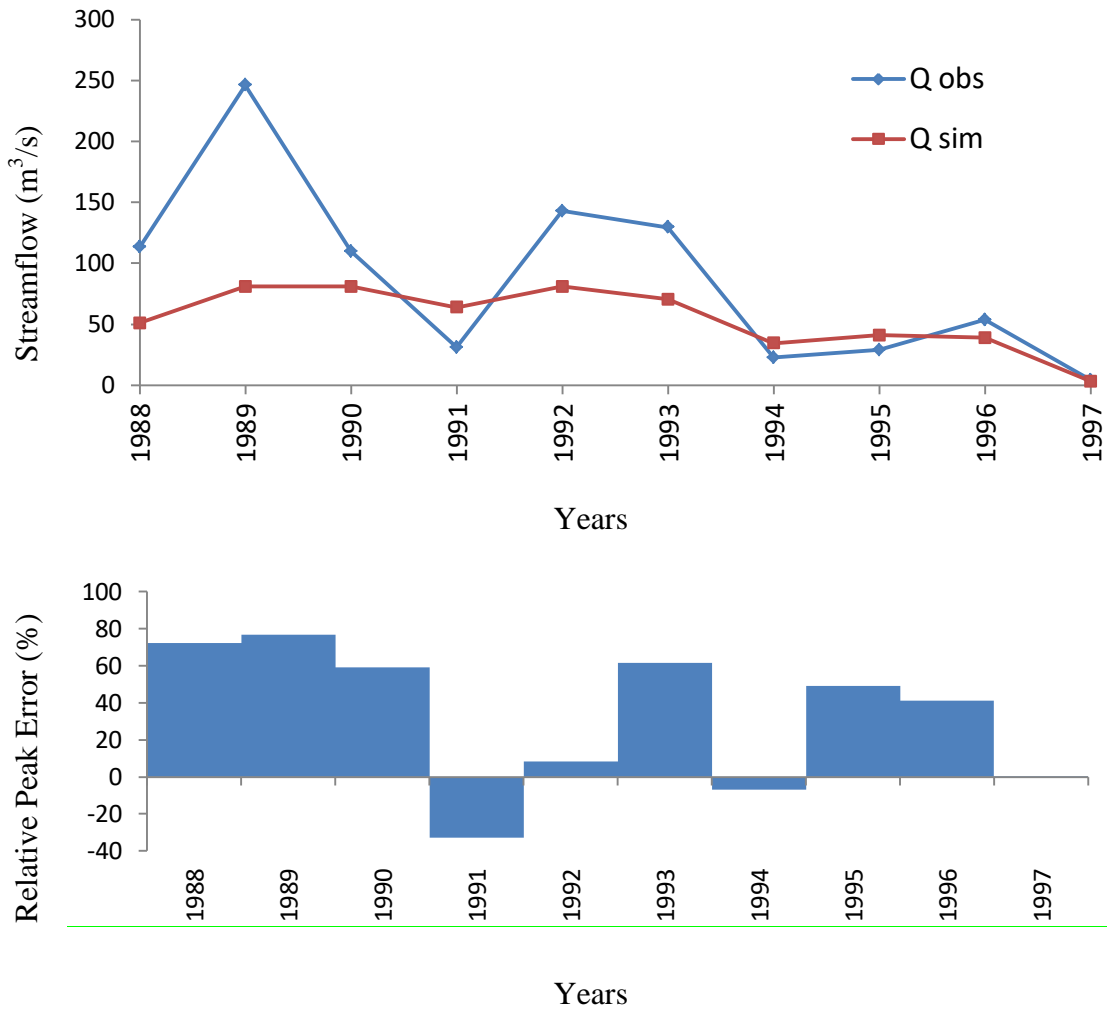


Figure.II.13. Peak flow estimate and relative peak error for FFNN model during testing phase.

II.9. Conclusion

The results obtained in this study showed the effectiveness of artificial intelligence algorithms for modeling the rainfall–runoff relationship for streamflow forecasting. Wavelet-feed forward neural network and adaptive neuro-fuzzy inference systems have a good predictive power. The performance of WFFNN and ANFIS in hydrological forecasting exceeds those of other models based on three different statistical indices (i.e., root mean squared error (RMSE), Nash-Sutcliffe efficiency (NSE), determination coefficient (R^2)) for training and testing

phases. The values of RMSE NSE and R^2 for the WFFNN model (RMSE = $3.772 \text{ m}^3 / \text{sec}$, NSE = 0.779 and $R^2 = 0.802$ for testing phase) were lower than those of ANFIS (RMSE = $3.992 \text{ m}^3 / \text{sec}$, NSE = 0.753 and $R^2 = 0.794$ for testing phase) models and FFNN (RMSE = $4.750 \text{ m}^3 / \text{sec}$, NSE = 0.649 and $R^2 = 0.630$ for testing phase). The use of these hybrid methods is an alternative fully justified for good water management and especially to minimize the risk of flooding within the watershed. In spite of these difficulties, modeling by WFFNN and ANFIS led to satisfactory results in forecasting the hydrological phenomena. This type of model represents a very powerful means for an estimated management of the surface water resources in a semi-arid to arid area particularly in the period of rise. These encouraging results open a number of perspectives; it would be interesting to try hybrid models by coupling wavelet transform with neuro-fuzzy systems.

Chapter. III

Applied Evolutionary Wavelet-ANFIS And Wavelet-FFNN On The Sebaou River.

III.1. Introduction

Sustainable development of human activities is based especially on an integrated management of water resources (Koudstaal, Rijsberman, & Savenije, 1992). Hence, an efficient and sustainable management of water resources cannot be limited to the simple guarantee of sufficient quantity and quality of water to meet the needs of humans (e.g. drinking, industry and irrigation) because it has to take into account the occurrence of extreme events, such as drought flow and flooding (Zakhrouf, Bouchelkia, & Stamboul, 2015). The complex nature of streamflow and the significant variability in spatial and temporal aspects have led to the development and application of stochastic concepts for modeling, forecasting, and other purposes (Martins, Sadeeq, & Ahaneku, 2011). Time series forecasting is one of the most and important methods used in hydrological modeling because it does not require understanding the internal structure of the physical processes (Krishna, Satyaji Rao, & Nayak, 2011). A time series is a sequence of regularly sampled quantities from an observed system, and a reliable time series prediction method can help researchers model the system and forecast its behaviors (Chen, Wang, Zou, Yuan, & Hou, 2014; Zhang, Chung, & Lo, 2008). Different mathematical methods have been suggested to solve time series forecasting, such as Linear Regression, Autoregressive Integrated Moving Average (ARIMA), and ARIMA with exogenous input. However, the complexity of the hydrological time series requires specific tools for non-linear and non-stationarity dynamic systems (Sivakumar, Berndtsson, Olsson, & Jinno, 2001). In recent years, the field of computational intelligence has promoted revolutionary changes in the development of new non-conventional techniques of data processing and simulation (Chandwani, Vyas, Agrawal, & Sharma, 2015). In this study, wavelet-based data-driven approaches are suggested for modeling the hydrologic time series. The combination of wavelet transform (WT) and artificial neural networks approaches has been successfully applied to

hydrological modeling and forecasting in the literature (Nourani, Baghanam, Adamowski, & Gebremichael, 2013). A wavelet-based artificial neural networks model is started from both WT and various artificial intelligence (AI) modeling techniques. Wavelet-based data-driven models, such as wavelet auto regressive moving average, wavelet artificial neural network (WANN), wavelet support vector regression, and wavelet adaptive neuro fuzzy inference system (WANFIS), have been applied as effective tools for forecasting complex and nonlinear hydrological time series (Kamruzzaman, Metcalfe, & Beecham, 2013; Liu, Niu, Wang, & Fan, 2014; Partal, 2009; Partal & Kisi, 2007; Seo, Kim, Kisi, & Singh, 2015; Seo, Kim, Kisi, Singh, & Parasuraman, 2016; Wang, Wang, Chen, Zhao, & Yang, 2013). Many studies have attempted to forecast time series of streamflow using wavelet-based artificial neural networks models (Badrzadeh, Sarukkalige, & Jayawardena, 2013; Guo, Zhou, Qin, Zou, & Li, 2011; Nourani, Baghanam, Adamowski, & Kisi, 2014; Tiwari & Chatterjee, 2010). The design of the optimal architecture of wavelet-based artificial neural networks models can be treated as a search problem in architectural space. In general, the architecture of artificial neural networks models is selected by trial and error without universal rules. Evolutionary algorithms (EAs) are a class of stochastic searches and optimization techniques obtained by natural selection and genetics. EAs mainly include evolutionary strategy, evolutionary programming, and a genetic algorithm (GA). The three EAs are consistent with the goal of using biological evolution mechanisms to improve the ability of using computers to solve the specific problems (Ding, Li, Su, Yu, & Jin, 2013). In this study, a method for designing wavelet-based artificial neural networks models, using GAs, was developed for forecasting streamflow. The performance of evolutionary wavelet-based artificial neural networks models has been investigated in several fields of engineering and science. The most recent studies using evolutionary wavelet-based artificial neural networks approaches for modeling hydrologic

systems are found in the literature (Asadi, Shahrabi, Abbaszadeh, & Tabanmehr, 2013; Danandeh Mehr, Kahya, & Olyaie, 2013; Kalteh, 2015; Kisi & Shiri, 2012; Ravansalar, Rajaei, & Zounemat-Kermani, 2016; Ravansalar, Rajaei, & Kisi, 2017; Sahay & Srivastava, 2014). The second section of this chapter provides the study area and data. The third suggests tools, including feed forward neural network (FFNN), adaptive neuro-fuzzy inference system (ANFIS) and GA. The fourth section explains the methodology, and the fifth presents results and discussion. Conclusions are found in the last section. This chapter includes partial contributions from the paper (Zakhrouf et al,2018).

III.2. Study area and data

Data derived from the Sebaou River were employed for calibration and testing all of the models developed in this study. The Sebaou River basin is located in the central part of northern Algeria (Algiers-Hodna-Soummam watershed), between 36.0 N and 37.0 N (latitude) and 3.0 E and 4.0 E (longitude). It covers a total catchment area of 871.54 km², with a perimeter of 248.5 km and compactness index equal 1.40 (Fig.III.1).

The region is characterized by a Mediterranean climate with cold winters and hot summers.

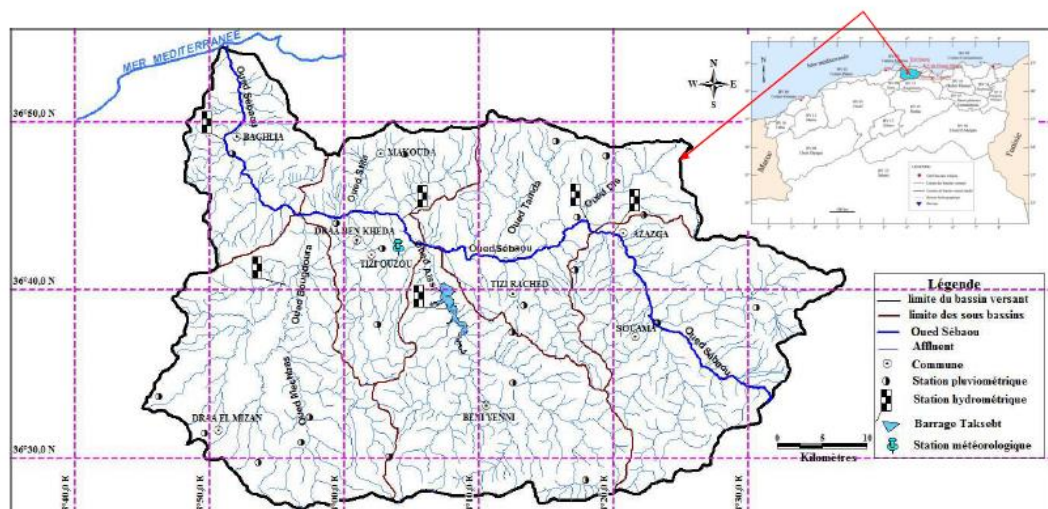


Figure. III.1. Sebaou watershed (Tarmoul et al, 2018).

Daily streamflow data, obtained from the National Agency of Water Resources (Algeria) of the Baghlia gaging station, were used in this study. Daily streamflow data of 14 years were used. The first 12 years (75% of the data) were used for training of the model, and the remaining 3 years (25% of the data) were used for testing (Fig.III.2).

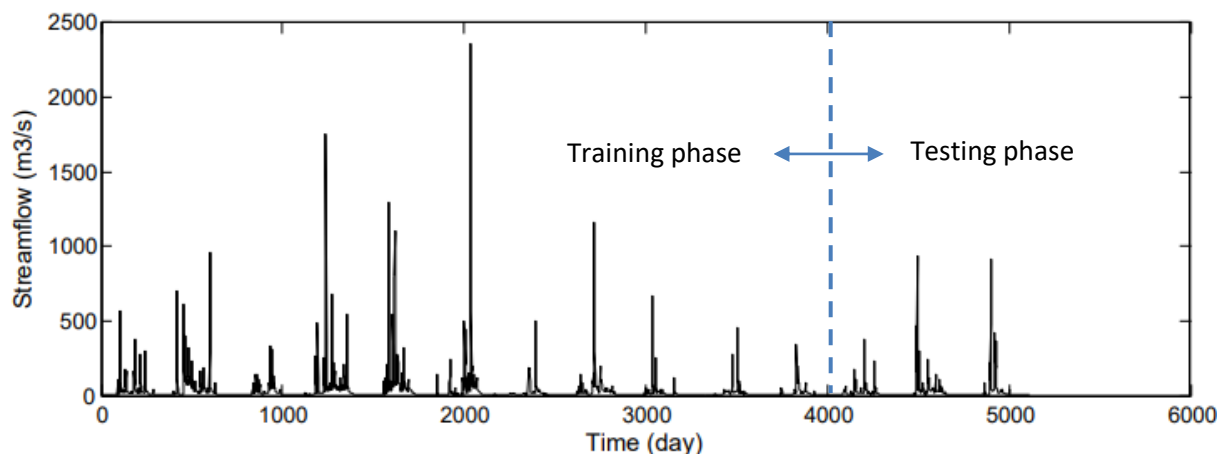


Figure. III.2. Average daily streamflow measured at Baghlia gaging station.

III.3. Materials and Methods

In this chapter we suggested the use of :Feed forward neural network (FFNN) (see chapter I), Adaptive Network-based Fuzzy Inference System (ANFIS) (see chapter I), Wavelet – FFNN and Wavelet – ANFIS (see chapter I) to forecast the streamflow of Sebaou river flow.

III.3.1. Genetic algorithm

Evolutionary algorithms [e.g., genetic algorithm (GA), evolutionary strategy (ES), and programming (EP)] are a class of stochastic searches and optimization techniques (Ding et al. 2013).

A GA is considered to be a heuristic and stochastic optimization technique based on evolutionary theory and genetic principles. GA has been for the most part techniques applied

by computer scientists and engineers to solve practical problems. However, Holland's (1975) original work on the subject was meant not only to develop adaptive computer systems for problem solving but also to shed light, via computer models, on the mechanisms of natural evolution (Dong, Wang, Sun, & Zhao, 2010).

Compared with other search techniques, GA is easy to use because it does not require derivative information or definite initial estimates in the solution space. Applying GA operators in the reproduction stage requires six components:

(1) Encoding,

Genetic algorithms traditionally use finite size binary, integer or real bit strings to represent chromosomes.

(2) Generating the initial population,

We define an initial population of individuals that corresponds to the first generation.

(3) Calculating the fitness values,

The fitness function measures the power of each chromosome to adapt, it describes the ability of a population chromosome to optimize the objective. This function is chosen according to the value of the objective function to be processed.

(4) Genetic operators,

- Selection options specify how the genetic algorithm chooses parents for the next generation.
- Crossover options specify how the genetic algorithm combines two individuals, or parents, to form a crossover child for the next generation.
- Mutation options specify how the genetic algorithm makes small random changes in the individuals in the population to create mutation children.

(5) Replacement,

Replacement the old solutions by the new best ones,

(6) stopping criteria.

III.4. Methodology

In this chapter, an effort has been made to use the combination model to forecast the daily streamflow of river located in the north of Algeria. In general, the time series models are used to describe the stochastic structure of the time sequence of a hydrological variable measured over time. In this case, the time series is decomposed into one comprising low frequencies and its trend (Q_A) and one comprising the high frequencies and the fast events ($Q_{D1}, Q_{D2}... Q_{Dn}$).

Decomposed streamflow time series were imposed using the previous times (lag) as inputs to the FFNN and ANFIS models for predicting flow of one day ahead. The sub-time series with lags are inputs and the original time series at time ($t + 1$) is output (Figure 5), which can be represented as:

$$Q(t) = FFNN/ANFIS [Q_A(t-t_1), Q_{D1}(t-t_2), Q_{D2}(t-t_3), \dots, Q_{Dn}(t-t_n)] \quad (III.1)$$

Where, $t_1, t_2, t_3...t_n =$ the lags of each sub-time series.

In parallel, GA is employed to find the best architecture and the best parameters of the WANFIS and WFFNN models. As a result of the combinations, four models (FFNN, WFFNN, ANFIS and WANFIS) have been developed and compared. The methodology for designing and training wavelet based artificial neural networks models (WFFNN and WANFIS) adopts a real coded GA strategy and hybrid with a back-propagation (BP) algorithm, by integrating the global searching advantage of GA and the local searching ability of BP algorithm.

GA is applied to optimize the initial topology of BP algorithm. Then, it utilizes the BP algorithm to train the artificial neural networks models more accurately. Using GA, each gene represents one parameter and a chromosome is constructed from a series of genes.

Accordingly, all the weights and biases of the WFFNN were codified. GA starts by randomly generating a set of genes. Using the standard genetic operations, the optimal solution that corresponds to the optimal structure of the model was obtained. FFNN and ANFIS training methods provide a non-linear mapping between inputs and outputs and are extremely useful in recognizing patterns in complex data. The idea is to use a fitness function that tests how well an architecture learns from the data (Fig.III.3). As regards to the fitness function, it is based on the root mean square error (RMSE) over a test dataset, which is represented by the following expression:

$$RMSE(C_j) = \sqrt{\sum_{i=1}^N (Qt_i - \hat{Q}t_i)^2 / N} \quad (III.2)$$

Where, Qt_i = the measured flow rate value and $\hat{Q}t_i$ = the flow rate calculated by the model obtained by testing dataset for j th chromosome $C(j)$.

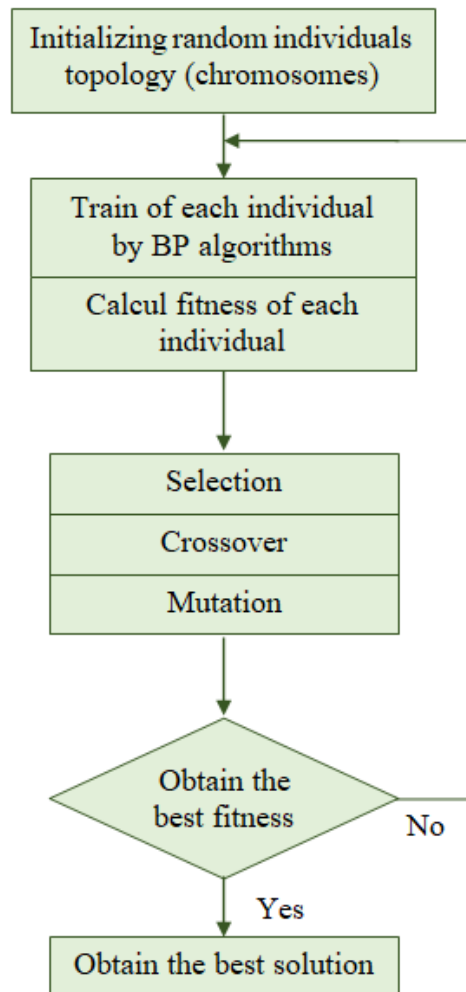


Figure. III.3. The flowchart of the optimization process of GA.

This research was focused on the combination between the methods of holdout cross validation and early stopping to calibrate the parameters of our models. Early stopping is a method for avoiding overfitting and requires the cross validation method to assess the relationship between the generalization accuracy of the learned model and the training accuracy. Holdout cross validation is one of the methods for estimating the generalization accuracy of a supervised learning algorithm (Prechelt., 1997; Arlot et al., 2010).

In this study, the calibration data (the 75% of the data) was separated into two sets, called the training and validation sets. The validation score can be used as a generalized measure. The

percentage of each set (training and validation) was determined by the GA to find the most appropriate validation values for generating the training performance of models. Then, the accuracy of the calibrated models was estimated using the testing (the remaining 25 % of the data) for the reliable confirmation. This study compares the effects of 20 selected wavelet functions on the performance of wavelet based artificial neural networks models. Wavelet functions were determined from the five most frequently used wavelet families. These five families are called as Haar (*Har*), Daubechies (*Db*), Coiflets (*Coif*), Symlets (*Sym*) and Biorthogonal (*Bior*), respectively. The effects of activation functions on the performance of FFNN was demonstrated by the following four selected activation functions including Linear (*Purelin*), Symmetric saturating linear (*Satlins*), Log-sigmoid (*Logsig*), and Hyperbolic tangent sigmoid (*Tansig*) transfer functions, respectively. For the membership function type, five membership functions (*MFs*) were investigated including Π -shaped (*Pimf*), Trapezoidal-shaped (*Trapmf*), Triangular-shaped (*Trimf*), Gaussian curve (*Gaussmf*), and Built-in Gaussian function (*Dsigmf*), respectively.

To find the optimal architecture for WFFNN and WANFIS, the standard codification of the parameters were applied for WFFNN and WANFIS. WFFNN applied the normal feed forward neural network that has five inputs and 20 neurons (maximum) in hidden layer, and one neuron in the output layer. A chromosome was constructed from a series of genes (as shown in Fig.III.4) to find the important parameters including type of mother wavelet, number of inputs and the lag of each input, dataset into validation, number of neurons, an activation function type in two layers, learning and momentum factor, the connexion input-output layers, the biases connexion, and the initial weight coefficients, respectively. In case of WANFIS, the important parameters can be divided as type of mother wavelet, number of inputs, the lag of each input, dataset into training and validation, number and type of membership functions,

learning and momentum factor, firing strength of a rule (min, max, prod, probor), and the definition of if-then rules (and/or) It can be created under a chromosome by a series of genes as shown in Fig(III.5).

<i>Genes</i>	<i>Description</i>	<i>Values</i>
g1	Mother wavelet type	{har, db, sym, coif, bior}
g2	Input 1 ; Lag t_1	{0,1} ; {1, 2, ..., 20}
g3	Input 2 ; Lag t_2	{0,1} ; {1, 2, ..., 20}
g4	Input 3 ; Lag t_3	{0,1} ; {1, 2, ..., 20}
g5	Input 4 ; Lag t_4	{0,1} ; {1, 2, ..., 20}
g6	Input 5 ; Lag t_5	{0,1} ; {1, 2, ..., 20}
g7	Validation data set	[5, 25]%
g8	Number of neurones	{2,3, ..., 20}
g9	Activation functions in hidden layer	{satlins, purelin, logsig, tansig, radbas}
g10	Activation functions in output layer	{ satlins, purelin, logsig, tansig}
g11	Learning factor	[0 1]
g12	Momentum factor	[0 1]
g13	Connexion input-output layers	{0,1}
g14	Biases connexion in hidden layer	{0,1}
g15	Biases connexion in output layer	{0,1}
g16		[0 1]
g17		[0 1]
⋮	Intial weight and biase coefficients	⋮
Gn		[0 1]

Figure III.4. Chromosome encoding of WFFNN.

The performances of WFFNN and WANFIS for streamflow forecasting were compared those of FFNN and ANFIS. FFNN and ANFIS used original series as input data of the models without decomposition by WT. FFNN and ANFIS models were applied to forecast the streamflow using the previous flows as inputs, which can be represented as:

$$Q(t) = FFNN/ANFIS [Q(t-t_1), Q(t-t_2), \dots, Q(t-t_n)] \quad (III.3)$$

Where, $t_1, t_2, t_3 \dots t_n$ = the lag time series of previous flow.

Genes	Description	Values
g1	Mother wavelet type	{har, db, sym, coif, bior}
g2	Input 1 ; lag t_1	{0,1} ; {1, 2, ..., 20}
g3	Input 2 ; lag t_2	{0,1} ; {1, 2, ..., 20}
g4	Input 3 ; lag t_3	{0,1} ; {1, 2, ..., 20}
g5	Input 4 ; Lag t_4	{0,1} ; {1, 2, ..., 20}
g6	Input 5 ; lag t_5	{0,1} ; {1, 2, ..., 20}
g7	Validation data set	[5, 25]%
g8	Number of membership functions	{2,3}
g9	Type of membership functions	{pimf, trapmf, trimf, gaussmf}
g10	Learning factor	[0 1]
g11	Momentum factor	[0 1]
g12	Firing strength of a rule	{(prod, max) ; (prod, probor) ; (min, max) ; (min, probor)}
g13	Definition of <i>if-then</i> rules type	{ and, or }

Figure.III.5. Chromosome encoding of WANFIS.

To assess the performance of the models (WANFIS, WFFNN, ANFIS and FFNN) to forecast daily streamflow during the testing phase, several measures of accuracy were applied. The mean absolute relative error (*MARE*), the root mean square error (*RMSE*), the NASH efficiency coefficient (*NSE*), the correlation between observed and predicted streamflows was expressed by means of the correlation coefficient (*R*), and the coefficient of determination (R^2) describe the proportion of the total variance in the observed data that can be explained by the models.

$$MARE = \frac{\sum_{i=1}^N \left| \frac{Qt_i - \hat{Q}t_i}{Qt_i} \right|}{N} \cdot 100 \quad (III.4)$$

$$RMSE = \sqrt{\sum_{i=1}^N (Qt_i - \hat{Q}t_i)^2 / N} \quad (III.5)$$

$$NSE = \left(1 - \frac{\sum_{i=1}^N (Qt_i - \hat{Q}t_i)^2}{\sum_{i=1}^N (Qt_i - \bar{Q}t)^2} \right) \cdot 100 \quad (III.6)$$

$$R = \frac{\sum_{i=1}^N (Qt_i - \bar{Q}t)(\hat{Q}t_i - \bar{Q}t)}{\sqrt{\sum_{i=1}^N (Qt_i - \bar{Q}t)^2 \sum_{i=1}^N (\hat{Q}t_i - \bar{Q}t)^2}} \quad (\text{III.7})$$

$$R^2 = \left(\frac{\sum_{i=1}^N (Qt_i - \bar{Q}t)(\hat{Q}t_i - \bar{Q}t)}{\sqrt{\sum_{i=1}^N (Qt_i - \bar{Q}t)^2 \sum_{i=1}^N (\hat{Q}t_i - \bar{Q}t)^2}} \right)^2 \quad (\text{III.8})$$

Where, Qt_i = the measured flow rate value, $\hat{Q}t_i$ = the flow rate calculated by the model, $\bar{Q}t$ = the average flow measured, $\bar{Q}t$ = the average flow simulated and N = the number of data

III.5. Results and application

The best structures and the optimal parameters of the four models were reported in Tables (III.1) and (III.2), respectively. Table (III.1) can explain that the best structure of FFNN corresponds to: (i) two input variables ($Q(t-1)$, $Q(t-11)$) that are the streamflow at one ($t-1$) and eleven ($t-11$) previous day; (ii) seventeen neurons in the unique hidden layer; and (iii) partitioning the data set into training, validation and test with 55%, 20% and 25% provided *the best results*. Furthermore, the best structure of the WFFNN corresponds to: (i) four input variables ($QD_1(t-1)$, $QD_2(t-1)$, $QD_4(t-17)$, $QA(t-1)$); (ii) thirteen neurons in the unique hidden layer; (iii) 'sym3' as a wavelet mother function; and (iv) partitioning the data set into training, validation and test with 60%, 15% and 25% provided the best results.

Table. III.1. Optimal structure of FFNN and WFFNN obtained by GA.

FFNN		WFFNN	
Input	$Q(t-1), Q(t-11)$	Input	$QD_1(t-1), QD_2(t-1), QD_4(t-17), QA(t-1)$
Conexion input-output	No	Conexion input-output	No
Hidden bias	Yes	Hidden bias	No
Output bias	No	Output bias	No
Number of neurons H-L	17	Number of neurons H-L	13
Activation function H-L	Tansig	Activation function H-L	Radbas
Activation function O-L	Satins	Activation function O-L	Purelin
Learning factor	0,5186	Learning factor	0.1936
Momentum factor	0,8282	Momentum factor	0.1108
Training set	55%	Training set	60 %
Validation set	20%	Validation set	15%
Testing set	25%	Testing set	25%
		Wavelet mother	'sym3'

The best structure of ANFIS and WANFIS were suggested in Table (III.2). It can conclude that the best structure of ANFIS corresponds to: (i) two input variables ($Q(t-1), Q(t-8)$) that are the streamflow at one ($t-1$) and eight ($t-8$) previous day; (ii) three membership functions; (iii) triangular membership functions; and (iv) partitioning the data set into training, validation and test with 70%, 5% and 25% provided the best results. Regarding the WANFIS, the best structure corresponds to: (i) four input variables ($QD_1(t-4), QD_2(t-2), QD_3(t-1), QA(t-1)$); (ii) two membership function; (iii) 'sym6' as a wavelet mother function; and (iv) partitioning the data set into training, validation and test with 70%, 5% and 25% provided the best results.

Table. III.2. Optimal structure of ANFIS and WANFIS obtained by GA.

ANFIS		WANFIS	
Input	$Q(t-1), Q(t-8)$.	Input	$QD_1(t-4), QD_2(t-2), QD_3(t-1), QA(t-1)$.
Number of membership functions	3	Number of membership functions	2
Membership functions	<i>Trimf</i>	Membership functions	<i>Gaussmf</i>
Definition of <i>if-then</i> rules (and/or)	<i>or rules</i>	Definition of <i>if-then</i> rules (and/or)	<i>and rules</i>
Firing strength of a rule	<i>or method = max</i>	Firing strength of a rule	<i>and method = min</i>
Learning factor	0.3878	Learning factor	0.1819
Momentum factor	0.9217	Momentum factor	0.0498
Training set	70%	Training set	70%
Validation set	5%	Validation set	5%
Testing set	25%	Testing set	25%
		Wavelet mother	'sym6'

Results in testing phase for FFNN, WFFNN, ANFIS and WANFIS models were shown in Table (III.3). According to Table (III.3), it is apparent that all the models provide different accuracies for different input combinations. It is clear from the Table (III.3) that the best results are obtained using WANFIS models, with $R = 0.934$ and $NSE = 0.87\%$; followed by the WFFNN; the FFNN in the third place and the ANFIS is the worst model. In terms of RMSE and MARE, the best performances were obtained using WANFIS and WFFNN at the same time, with $RMSE = 12.15 \text{ m}^3/\text{sec}$ and $15.72 \text{ m}^3/\text{sec}$, respectively. It can be imagined from results that WANFIS model can be chosen as the best model in this study.

Table. III.3. Comparison between the performance results obtained by the models: WANFIS, ANFIS, WFFNN, and FFNN in the testing phase.

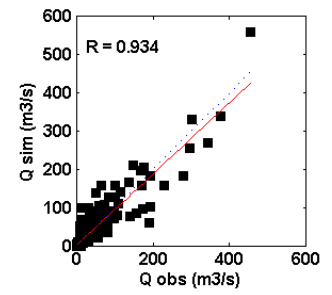
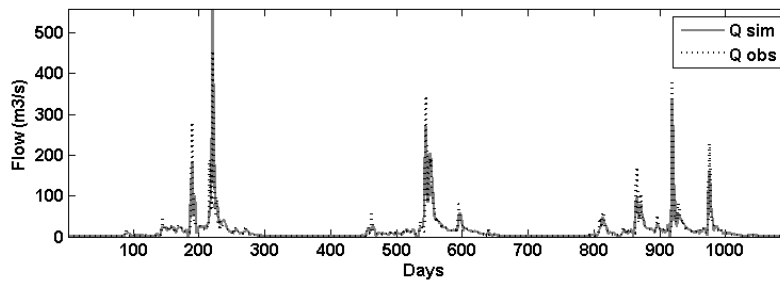
	RMSE m ³ /s	MARE %	R	R ²	NSE
WANFIS	12.1545	106.785	0.934	0.872	0.873
ANFIS	23.1326	324.413	0.748	0.560	0.541
WFFNN	15.7299	209.706	0.888	0.789	0.788
FFNN	22.4265	140.841	0.755	0.570	0.569

Figure (III.6) expresses the hydrography and scatter plot for WANFIS and ANFIS, during the testing phase. And, figure (III.7) represents the hydrography and scatter plot for WFFNN and FFNN during the testing phase. It was observed that the flow forecasted by WANFIS model was close to the 45° line. From this analysis, it was worth to mention that the performance of WANFIS was much better than other models. The magnitude of systematic over estimation or under estimation of a model is evaluated using the error and relative error indices.

Figure (III.8) presents the distribution of error and relative error along with the magnitude of streamflow computed by WANFIS and ANFIS during the testing phase.

Additionally, figure (III.9) shows the distribution of error and relative error along with the magnitude of streamflow, computed by WFFNN and FFNN during the testing phase. The figures provided that WANFIS forecasted the streamflow approximate to the general behavior of observed data. In particular, the peaks of the testing period were satisfactorily estimated by the WANFIS. The performance of WANFIS in the forecasting of peak flow was superior to the WFFNN and also to the conventional ANFIS and FFNN. Although WFFNN forecasted the streamflow close to the general behavior of observed data, peak flow could not be estimated satisfactorily by the WFFNN.

(a)



(b)

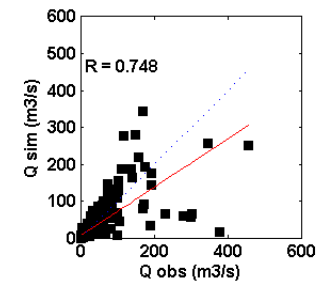
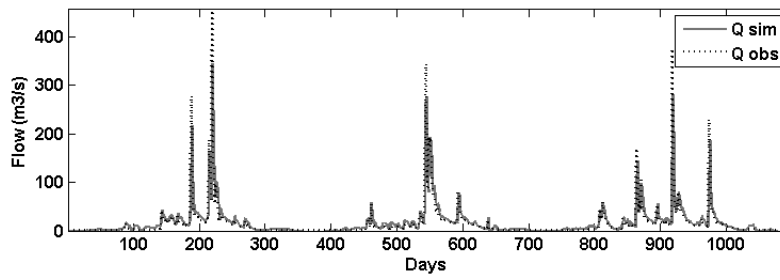


Figure. III.6. Hydrograph and scatter plot in the testing phase (a) WANFIS, (b) ANFIS.

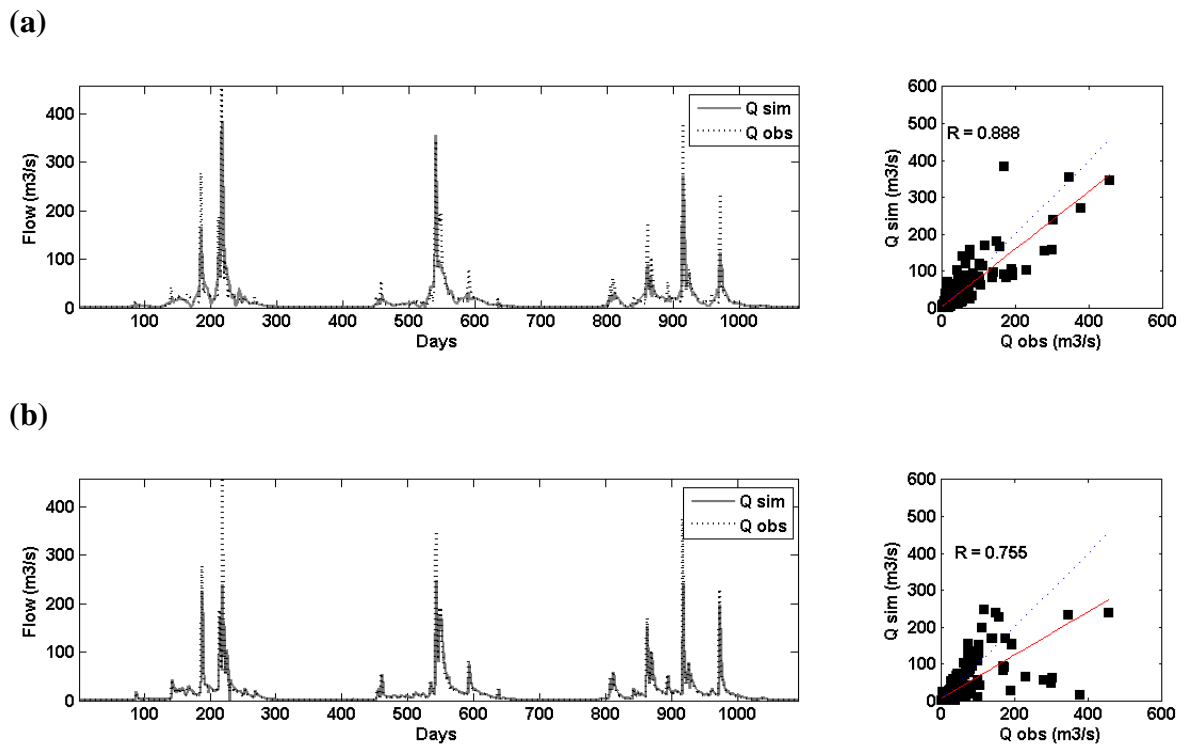
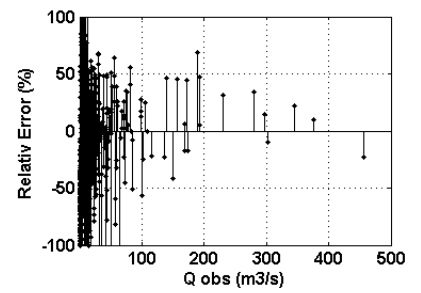
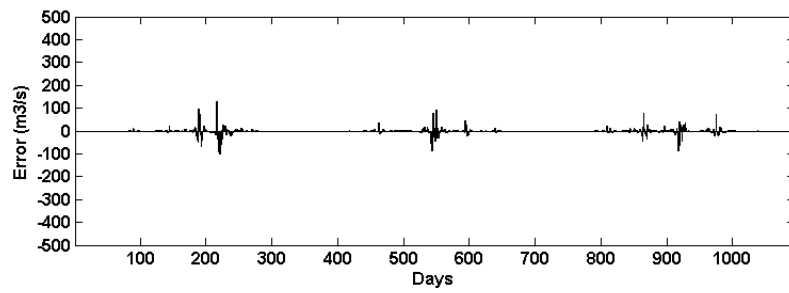


Figure. III.7. Hydrograph and scatter plot in the testing phase (a) WFFNN, (b) FFNN.

The reason for the better performance of WFFNN and WANFIS can be expressed that WFFNN and WANFIS used sub-time series as input data of the models, while FFNN and ANFIS models used original series as input data of the models without decomposition. Sub-time series can capture the high variations that existed in the original series, especially help to understand and capture the seasonality effect which include various information about the studied phenomenon, also wavelet transformation is reducing the noise in the streamflow time series causing forecasts to be more reliable and accurate (Tiwari and Chatterjee, 2010).

(a)



(b)

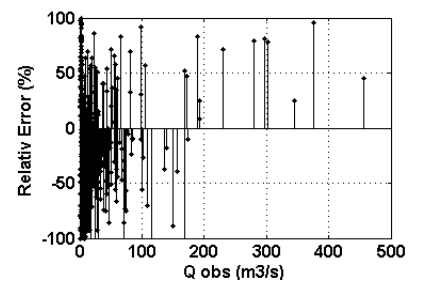
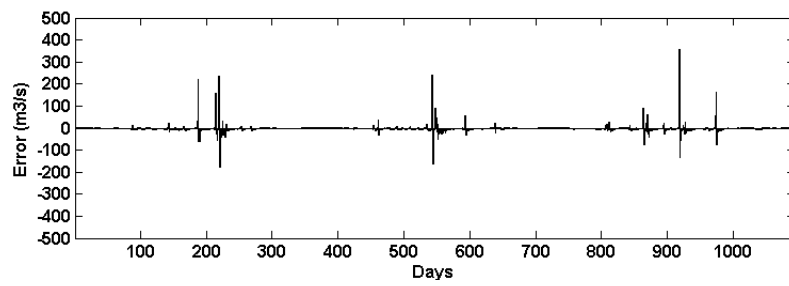
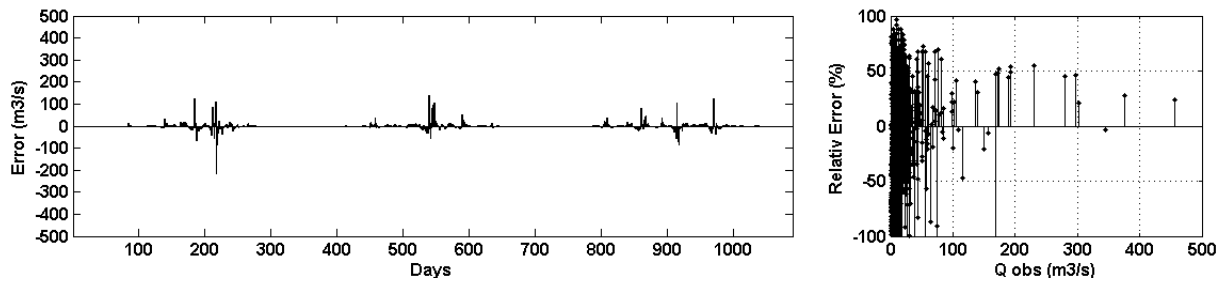


Figure. III.8. Error and relative error in the testing phase (a) WANFIS, (b) ANFIS.

(a)



(b)

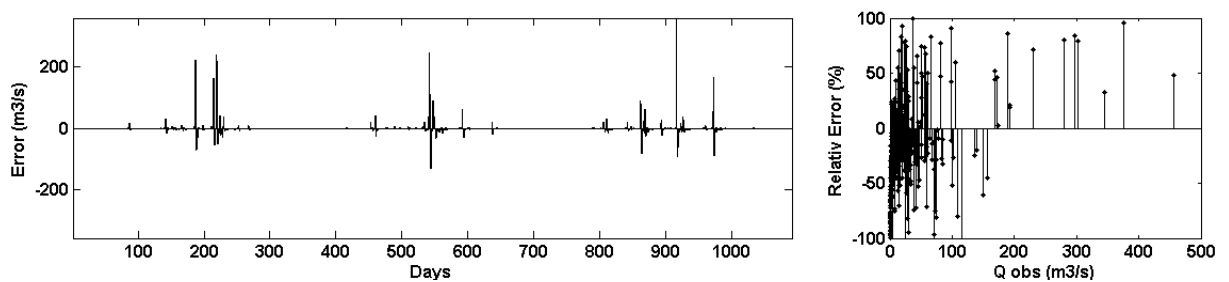


Figure. III.9. Error and relative error in the testing phase (a) WFFNN, (b) FFNN.

III.6. Conclusion

In this work, wavelet based feed forward neural network (WFFNN) and wavelet based adaptive neuro-fuzzy inference system (WANFIS) models using genetic algorithm (GA) are developed and used for forecasting the streamflows using the previous values of time series. Wavelet transform (WT) decomposes the original series into a sub-time series at different level. WFFNN and WANFIS models were also compared with the conventional feed forward neural network (FFNN) and adaptive neuro-fuzzy inference system (ANFIS) models, respectively. The forecasting accuracy of combination models (i.e. WANFIS and WFFNN) are significantly superior to the ones obtained by the conventional models (i.e. ANFIS and FFNN).

Sub-time series decomposed by WT as input data of the conventional models can improve the performance of the FFNN and ANFIS.

The accuracy and effectiveness of developed models (i.e. WFFNN, WANFIS, and FFNN, ANFIS) are evaluated based on four statistical indices, including root mean square error (RMSE), Nash-Sutcliffe efficiency (NSE), and correlation coefficient (R).

WANFIS (RMSE = 12.15 m³/s, NSE = 0.873, R = .934) and WFFNN (RMSE = 15.73 m³/s, NSE = 0.788, R = .888) models improved the performances of ANFIS (RMSE = 23.13 m³ /s, NSE = 0.541, R = .748) and FFNN (RMSE = 22.43 m³ /s, NSE = 0.569, R = .755) during the test period.

WANFIS model presents a good fit to the observed data, especially for the peak values in the testing period.

This result is quite significant since the conventional models can face the difficulties for forecasting the daily extreme values of the observed streamflow. The results obtained in this study expresses the effectiveness of combinations (i.e. wavelet transformation, artificial neural networks models and genetic algorithm) to forecast the daily streamflow. These encouraging results open a number of perspectives; it would be interesting to try hybrid models by coupling wavelet transform with other artificial intelligence models, simultaneously optimizing by genetic algorithm, and comparing the performance of our models with wavelet-genetic programming model.

Chapter. IV

Applied Evolutionary Recurrent Neural Networks. A case study: Chellif and Soummam Watersheds.

IV.1. Introduction

Streamflow forecasting and modeling procedures are the important and fundamental activity of hydrological sciences and engineering field, and define a primary non-structural category of flood prevention or water resource management (Seo et al., 2015a; Zakhrouf et al., 2018; 2020). Several researchers, engineers, professor, and hydrologists are frequent users of mathematical techniques based on stochastic and statistical processes because of nonlinear complexities of diverse events including streamflow, rainfall, evaporation, water stage, groundwater table, and lake inflow etc. (Lawrance and Kottegoda, 1977). In the past decades, time series analysis has received the considerable interest from many researchers in various fields including hydrological domain. Time series models are more simple than the complex mathematical and conceptual models because it does not require the internal relationship of physical procedures when the modeling frameworks are necessary (Salas, 1993). Streamflow forecasting, one of the popular time series techniques, has utilized the physical, statistical, and stochastic methods. Also, the conventional machine learning methods have proved that the model performance is quickly degraded when the data available include a unexplainable distortion and ambient noise (Le et al., 2019). Therefore, the significant researches have been investigated to improve the accuracy of time series models. Artificial neural network (ANN) has received a great deal of popularity under the banner of machine learning. Machine learning models including feedforward connection (e.g., feedforward neural network (FFNN)) are the most popular architectures. Recently, researches utilizing deep neural network strategies for time series analysis have suggested and displayed the promising results. In particular, the Elman recurrent neural network (ERNN), long short-term memory (LSTM), and gated recurrent unit (GRU) models have succeeded in solving the complex problems such as energy demand forecasting (Torres et al., 2018; Gökgöz and Filiz, 2018), speech

recognition (Cheng et al., 2019; Shewalkar et al., 2019), financial and stock market forecast (Sethia and Raut, 2019; Qiu et al., 2020), wind speed forecasting (Khodayar et al., 2017; Chen et al., 2019). Also, they have been tested successfully for overcoming the potential weakness of streamflow forecasting (Fu et al., 2020; Kimura et al., 2020; Liu et al., 2020; Zuo et al., 2020; Zhu et al., 2020). The performance of evolutionary optimization methods (e.g., ant colony (AC), genetic algorithm (GA), and particle swarm optimization (PSO) etc.) has been investigated to optimize the parameters and hyperparameters of artificial neural networks models in various fields (Baldominos et al., 2018; Fielding and Zhang, 2018; Elmasry et al., 2020; Zakhrouf et al., 2020). Because of simple implementation and higher capacity to find the global optimization value, the evolutionary optimization methods are recommended. The PSO is a computational method that optimizes the solution of addressed problems by using on a population of candidate particles and pushing these particles around in the search-space for the location and velocity of best solution. The key contribution of this work is to forecast the streamflow focused on an approach that combines the PSO to detect the most appropriate artificial neural networks components and correct input lag-times. In this chapter, four artificial neural networks (i.e., FFNN, ERNN, LSTM, and GRU) models are applied to forecast the streamflow with lag-times at Sidi Aich and Ponteba Defluent stations, Algeria. To obtain the optimal parameters and hyperparameters of applied models, adaptive moment estimation (ADAM) algorithm, an extension of stochastic gradient descent that has recently been widely implemented for the artificial neural networks approaches, is utilized to train the networks. The rest of the chapter is arranged as follows: section II arrays the materials and methods. The research area and dataset analysis are shown in the section III. The results and discussion are presented in section IV. Conclusion is followed in the last part of chapter.

IV. 2. Materials and Methods

In this chapter, four artificial neural networks (i.e. FFNN, ERNN, LSTM, and GRU) models are used (see chapter I for more detail). Also, the adaptive moment estimation (ADAM) algorithm is applied for model training and particle swarm optimization (PSO) is utilized for parameters and hyperparameters selection.

IV.3. Parameters and Hyperparameters Tunning

IV.3.1 Adaptive moment estimation (ADAM) algorithm

Backpropagation (BP) and backpropagation through time (BPTT) algorithms are one of the easiest ways for controlling training procedure in the machine learning. BP allows the users to calculate an effective error for each hidden layer based on a derivative method. Adaptive moment estimation (ADAM), a process of optimization to minimize the cost function, optimization algorithm is one of the most popular gradient descent optimization algorithms within artificial neural networks category because it produces accurate and fast results (Kingma and Ba, 2014; Wang et al., 2019). With the ADAM algorithm, an error function measuring the difference between observed (y_k) and expected (d_k) outputs can be minimized as follow:

$$E(w, x, d_k) = \frac{1}{2} \sum_1^k (d_k - y_k)^2 \quad (IV.1)$$

ADAM algorithm involves two steps: (1) a "forward move" during which network outputs are estimated from inputs and (2) a "backward move" during which partial derivatives of a certain cost function E with respect to parameters are replicated. Finally, the weights are modified according to the following equations:

$$\Delta w_{ij}(n) = -\varepsilon \frac{\partial E}{\partial w_{ij}}(n) + \alpha \Delta w_{ij}(n-1) \quad (IV.2)$$

$$w_{ij}^n = w_{ij}^{n-1} - \alpha_{n-1} \cdot \frac{\sqrt{1-\beta_2^n}}{\sqrt{1-\beta_1^n}} \cdot \frac{m_{ij}^{n-1}}{\sqrt{\vartheta_{ij}^{n-1} + \varepsilon}} \quad (\text{IV.3})$$

where,

$$\alpha_{n-1} = \frac{\alpha}{\sqrt{n}} \quad (\text{IV.4})$$

$$m_{ij}^{n-1} = \beta_1 m_{ij}^{n-2} + (1 - \beta_1) \frac{\partial E(w, x, d_k)}{\partial w_{ij}^{n-1}} \quad (\text{IV.5})$$

$$\vartheta_{ij}^{n-1} = \beta_2 \vartheta_{ij}^{n-2} + (1 - \beta_2) \left[\frac{\partial E(w, x, d_k)}{\partial w_{ij}^{n-1}} \right]^2 \quad (\text{IV.6})$$

Where α = the learning rate, w = the weights, ∂E and ∂w_{ij} = the partial derivatives of error and weights for each layer, β_1 and β_2 = first-order and second-order momentum, and n = the iterations.

IV.3.2. Particle swarm optimization (PSO) optimization algorithm

Evolutionary optimization algorithm inspired from the observation of natural behavior is one of popular schemes to determine the optimized hyperparameters and parameters for artificial neural networks models. It usually provide an efficient solution (Bouktif et al., 2020).

Particle swarm optimization (PSO) optimization algorithm is developed to imitate the social actions of birds and fish first suggested by Eberhart and Kennedy (1995). Compared with other metaheuristic and evolutionary optimization algorithm, PSO has two important advantages and properties. First, PSO algorithm is easily implemented. Second, it can keep a memory of good solution until the global solution of problem is found (Seo et al, 2016; Hu et al., 2019).

The swarm of particles that defined by a sets of vectors of N dimension, can be considered as a candidate solution of the problem. Every particle in the swarm has three compound vectors. (1) the position (P) identifies the current position of that particle, (2) the velocity (V) determines direction and speed of that particle, and (3) the personal best (Pbest) indicates the best position

of particle between all swarms. Also, the global best (Gbest), a significant swarm vector, stores the best location tested over the swarm.

Each particle is evaluated using an iterative method, so the personal best (Pbest) for each particle and the global best (Gbest) for the swarm are updated at the end of each iteration. For each iteration, the position (P) and velocity (V) of particle are modified using the following equations:

$$V_i^{t+1} = V_i^t + rand(0,1). (P_i^{best} - P_i^t) + rand(0,1). (G_{best} - P_i^t) \quad (IV.7)$$

$$P_i^{t+1} = P_i^t + V_i^{t+1} \quad (IV.8)$$

where $rand(0,1)$ = the random values uniformly distributed in $[0,1]$.

IV.4. Research area and data analysis

IV.4.1. Soummam and Chellif River Watersheds

The Soummam watershed is located in the northeastern Algeria and bordered by the Mediterranean Sea (North), Djurdjura (West), coastal massifs of Bejaia (Northwestern), high plains of Sétifien (East), and Bouira plateau (South), respectively. It is drained by the Bou-Sellam and the Sahel rivers which form the Soummam River. Also, it extends over an area of 9125 Km² based on the longitudes (between 3°60' E and 5°57' E) and latitudes (between 35°75' N and 36°77' N). In addition, the perimeter and compactness index are 554 km and 1.62 (Fig. IV.1).

The Cellif River watershed is located in the northwestern Algeria and bounded by the Mediterranean Sea (North), Oranie-Chott-Chergui region (West), Desert (South), and Algerian Chott-Hodna region (East). The area of Cellif River watershed is 43700 km² based on the longitudes (between 0°0' E and 3°5' E) and latitudes (between 33°5' N and 36°5' N). Also, 1383 km of perimeter and its compactness index is equal to 1.85 (Fig. IV.2).

The climates of two watersheds are under the influence of three climatic regimes including coastal temperature, Tellian Atlas, and High Plateau climates. The annual rainfall of two watersheds varies from 300mm (the Setif plateaus) to 1000mm (Coast).

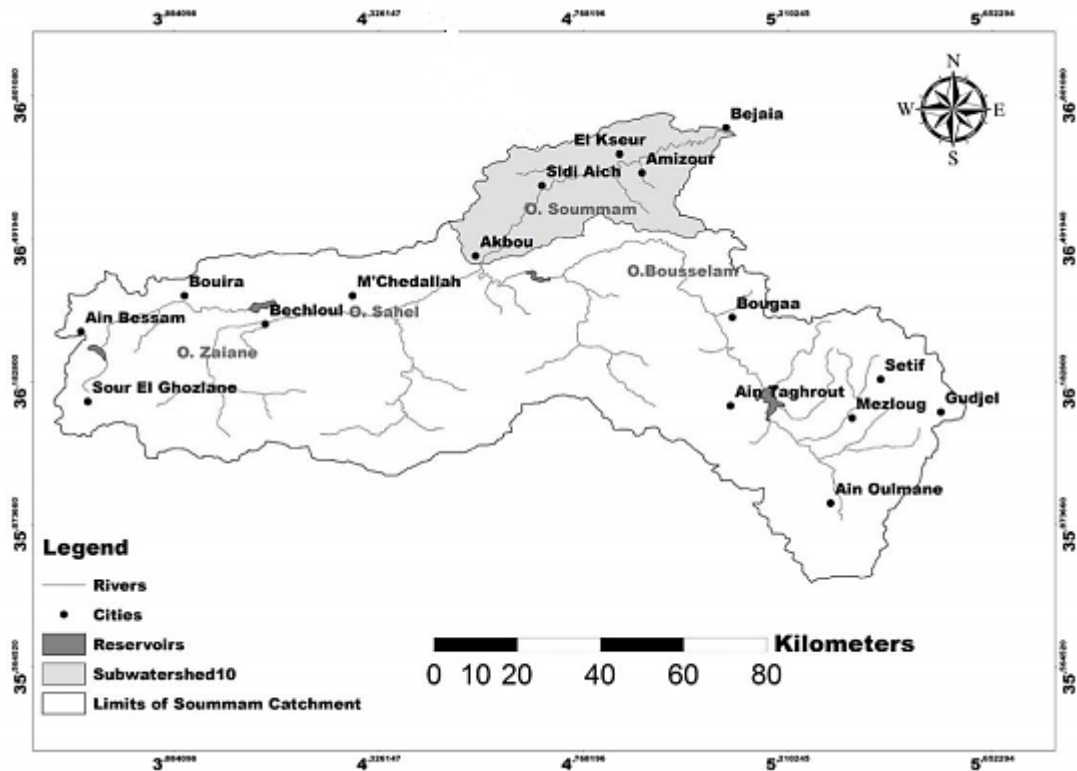


Figure. IV.1. Soummam watershed (Allili-Ailane et al, 2015).

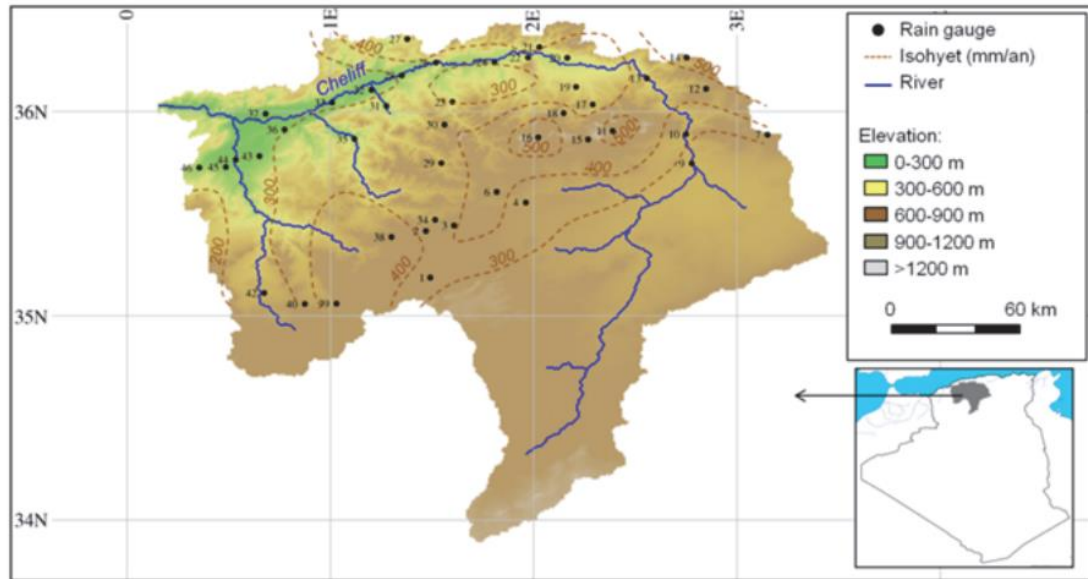


Figure. IV.2. Chellif watershed (Benhattab et al, 2011)

IV.4.2. ACF and PACF analysis

The purpose of this research is to evaluate the performance of artificial neural networks models including the feedforward neural network (FFNN), Elman recurrent neural network (ERNN), long short-term memory (LSTM), and gated recurrent unit (GRU) for forecasting one-day-ahead streamflow based on the previous streamflow.

A dataset containing the daily streamflow during six years were collected from Sidi Aich (Sommam watershed) and Ponteba Defluent (Chellif watershed) stations of the National Water Agency of Algeria. They were subdivided into training and testing groups, respectively. The first five year data (83% of the total dataset) was utilized for model training, and the last one year data (17% of the total dataset) was utilized for model testing. Figures (IV.3)-(IV.4) show the observed streamflow hydrograph for Sidi Aich and Ponteba Defluent stations.

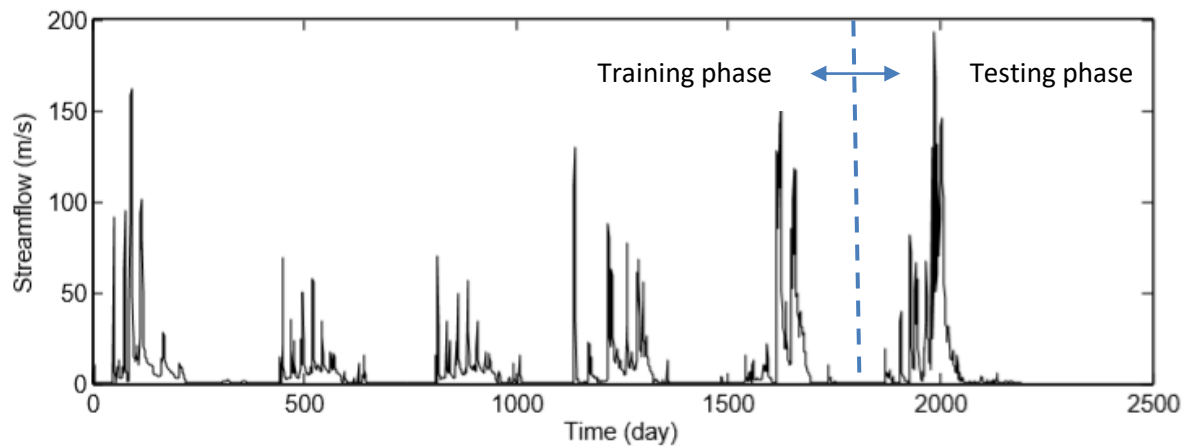


Figure. IV.3. Observed streamflow hydrograph (6years) of Ponteba Defluent station.

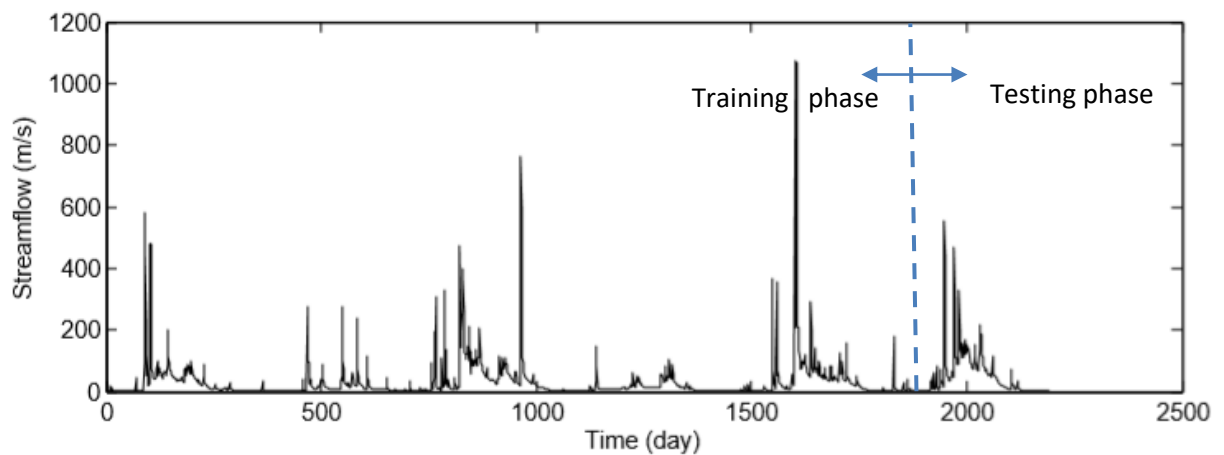


Figure. IV.4. Observed streamflow hydrograph (6years) of Sidi Aich station.

The choice problem of appropriate lag-time is the most important task for the successful modeling (Seo et al., 2015b; Sudheer et al., 2002). Autocorrelation function (ACF) and partial autocorrelation function (PACF) are one of conventional techniques that utilized to define the best lag-time for forecasting model. ACF and PACF for the lag-times corresponding to 30 days and 95% confidence band for the Sidi Aich and Ponteba Defluent stations are provided in Figs (IV.5. a-b), (IV.6. a-b).

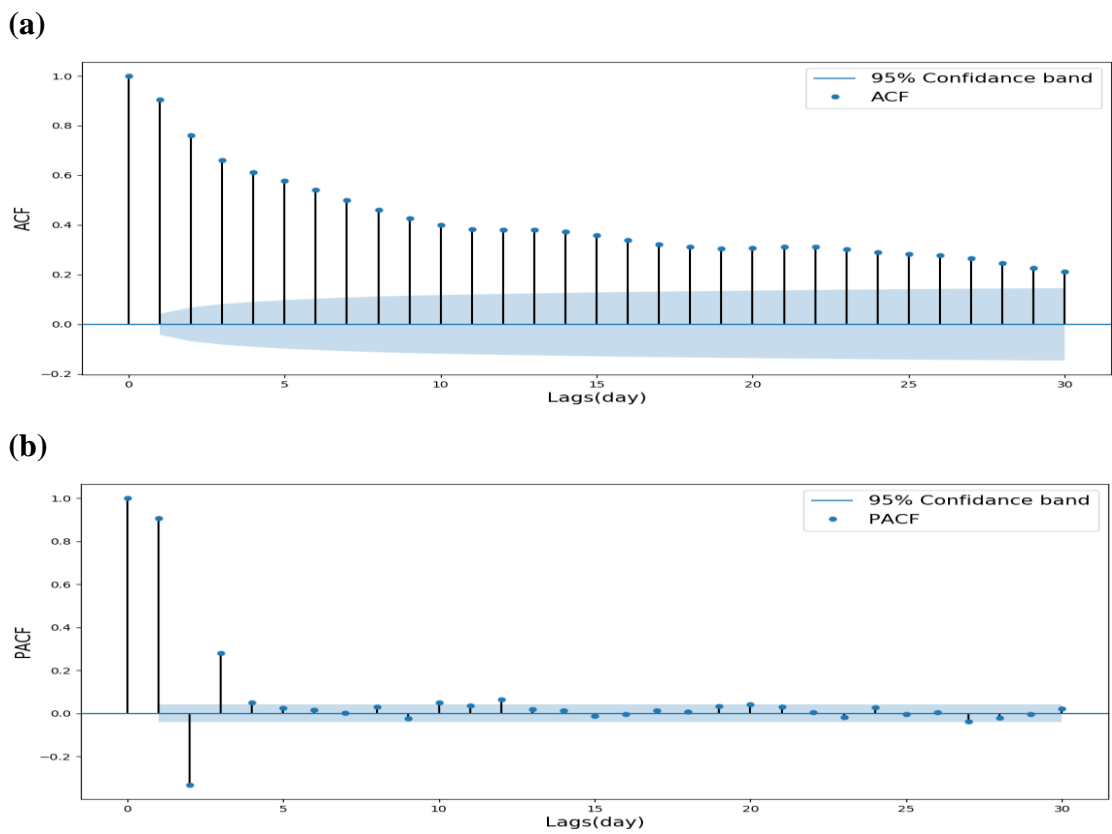


Figure. IV.5. a) Autocorrelation and b) partial autocorrelation correlograms for Ponteba Defluent streamflow time series.

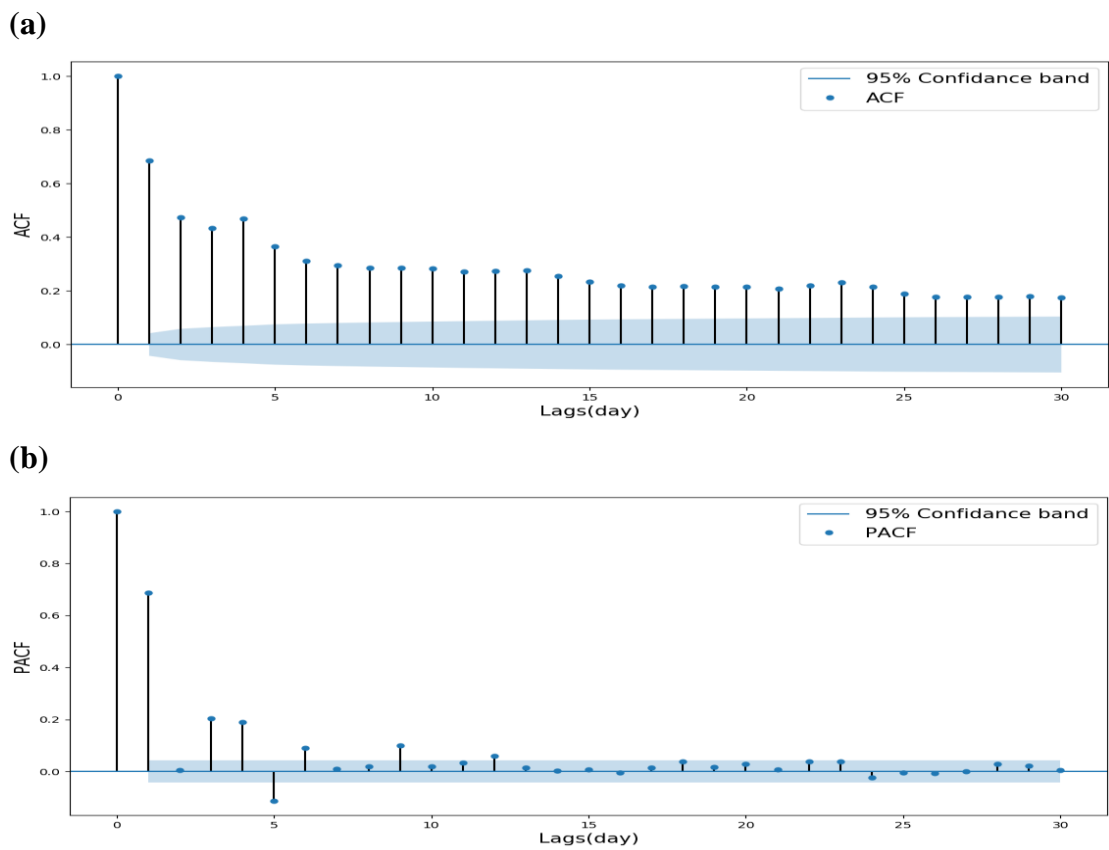


Figure. IV.6. a) Autocorrelation and b) partial autocorrelation correlograms for Sidi Aich streamflow time series.

Figures (IV.5. a-b) and (IV.6. a-b) indicated that the values of ACF rapidly decreased from 1 to 4 lag-times and gradually reduced after 5 lag-time at both stations. Also, the analysis of PACF values showed that the high values of 1, 3, and 4 lag-times provided a significant correlations by the dependence of successive variable at Sidi Aich station, whileas the high values of 1-3 lag-times at Ponteba Defluent station. Using the analysis of ACF and PACF correlograms, the lag-times were condisered from one-day-ahead to four-day-ahead for streamflow forecasting at both stations (Table. IV.1).

Table. IV.1. Model based inputs propositions.

Models	Inputs propositions
I	$Q_{t+1} = f(Q_{t-1})$
II	$Q_{t+1} = f(Q_{t-1}, Q_{t-2})$
III	$Q_{t+1} = f(Q_{t-1}, Q_{t-2}, Q_{t-3})$
IV	$Q_{t+1} = f(Q_{t-1}, Q_{t-2}, Q_{t-3}, Q_{t-4})$

IV.5. Statistical Indices

In this chapter, three statistical indices (i.e., root mean squart error (RMSE), signal-to-noise ratio (SNR), and Nash–Sutcliffe efficiency (NSE)) are hired to evaluate the forecasting accuracy and show the capacity and performance of different models. RMSE indice is frequently used to evaluate how closely the forecasted values match the observed ones based on the measure of square root of mean errors between the forecasted and observed values using equation (IV.9). SNR indice can be addressed as the ratio of meaning information to the undesirable one using equation (IV.10). If SNR indice equals to zero, it indicates the perfect performance between observed and forecasted streamfow. However, the higher value can be explains the poor performance (Bormann, 2005; Moriasi et al., 2007). NSE (ASCE, 1993;

Legates and McCabe, 1999) indice a normalized statistic that determines the relative magnitude of residual variance using equation (IV.11). NSE values providing close to zero indicates the unacceptable performance, Whereas NSE values showing near one considered as good performance.

$$RMSE = \sqrt{\frac{\sum_{i=1}^N (Q_{t_i} - \hat{Q}_{t_i})^2}{N}} \quad (IV.9)$$

$$NSR = \frac{\sqrt{\sum_{i=1}^N (Q_{t_i} - \hat{Q}_{t_i})^2 / \gamma}}{\delta} \quad (IV.10)$$

$$NASH = 1 - \frac{\sum_{i=1}^N (Q_{t_i} - \hat{Q}_{t_i})^2}{\sum_{i=1}^N (Q_{t_i} - \bar{Q}_t)^2} \quad (IV.11)$$

where Q_{t_i} = the observed streamflow value, \hat{Q}_{t_i} = the forecasted streamflow value, \bar{Q}_t = the mean observed streamflow, \tilde{Q}_t = the mean forecasted streamflow, N = the number of data, γ = the number of degrees of freedom, and δ = the standard deviation of observed streamflow.

IV.6. Implementation of developed models

Artificial neural networks models have become exceedingly popular for solving forecasting problems. Selecting an appropriate architecture can significantly affect model performance. When designing the artificial neural networks models, choosing the hidden structure (e.g., number of hidden layers, hidden nodes, and activation function type etc.) is the important priority before model training. Also there are another kind of hyperparameters related to the optimization algorithm that can ameliorate the model performance such as momentum and learning rate.

In this chapter, PSO optimization algorithm was hired to select the hyperparameters of artificial neural networks models. Also, ADAM optimization algorithm was utilized to train the model and determine the parameters such as weights and biases. The flowchart applying the models for streamflow forecasting in this work can be found in fig (IV.7).

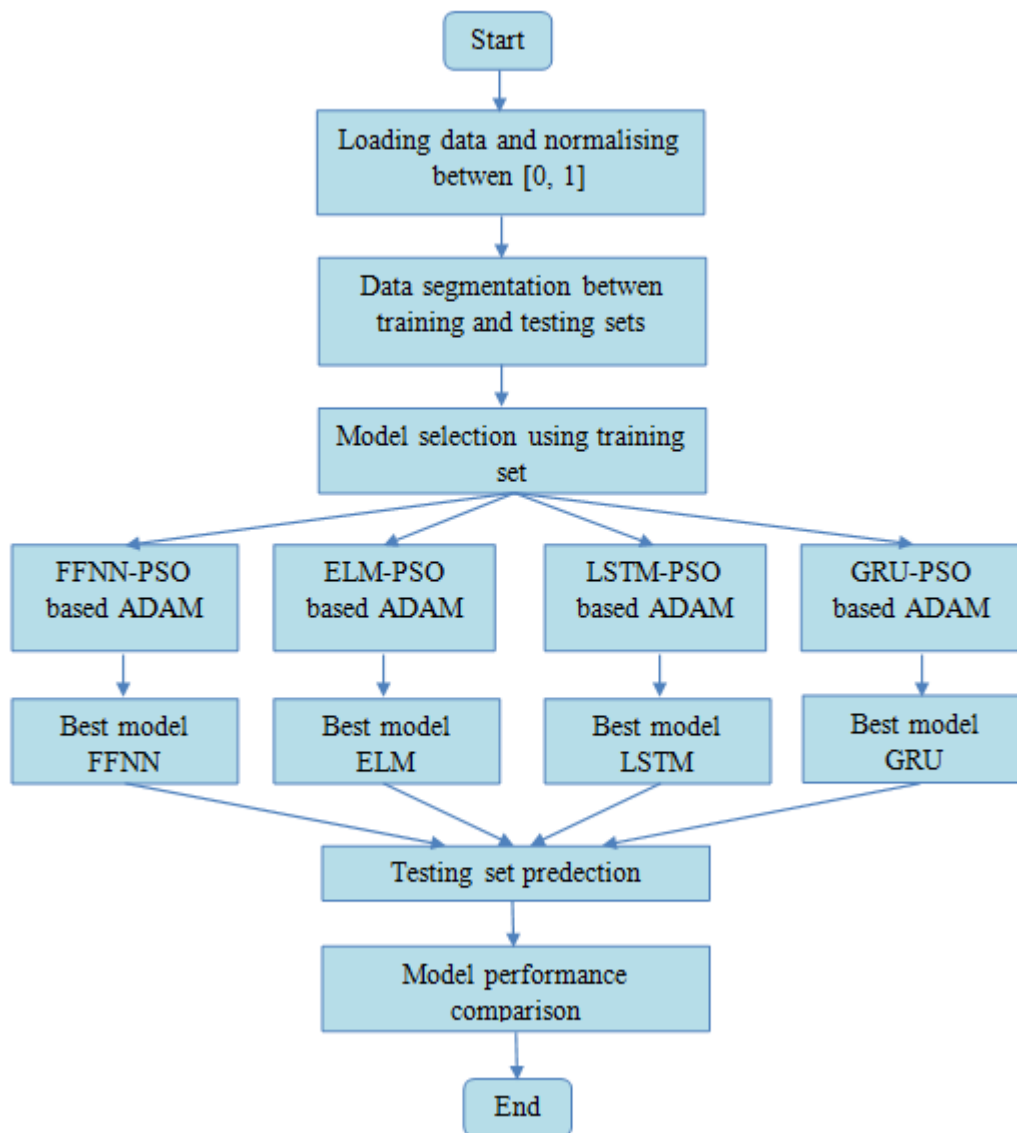


Figure. IV.7. The flow chart on applying the models in forecasting streamflow.

The research steps for this purposes can be described as follows:

[step1] Encoding: In this step, the hyperparameters need to be optimized are encoded into fixed-length vectors. For artificial neural networks models, there are eight key parameters including number of hidden layers (NHL), number of neurons in hidden layers (NNHL), activation function type in hidden layers (AFHL), activation function type in output layer (AFOL), learning rate (LR), first-order momentum (FOM), second-order momentum (SOM),

and the batch size (BS), respectively. The effects of activation functions on models performance demonstrated utilizing four activation functions including linear transfer function (Linear), rectified linear unit (ReLU), sigmoid transfer function (Sigmoid), and hyperbolic tangent sigmoid transfer function (Tansig), respectively. Table (IV.2) shows the defined variables and domains.

[step 2] Initializing randomly individual position and velocity.

[step 3] Training the artificial neural networks models using the ADAM algorithm for each particle, and calculate the fitness score of each model. In this chapter, each optimized model will be chosen based on NSE statistical indice as fitness function of PSO optimization algorithm.

[step 4] Update personal best (Pbest) for each particle and the global best (Gbest) of swarm.

[step 5] Update position (P) and velocity (V) for each particle utilizing equations (IV.7) and (IV.8).

[step 6] Stopping criteria; otherwise, go to step 3.

Table. IV.2. Variables encoded and their default domains.

Hyperparameter	Interval values	Type
Input model based lags (IM)*	Model{I, II...IV}	Discret
Number of hidden layer (NHL)	{1,2...6}	Discret
Number of neurones in hidden layers (NNHL)	{2,3,...200}	Discret
Activation function type in hidden layer (AFHL)	{ ReLU, Tansig, Sigmoide }	Discret
Activation function type in output layer (AFOL)	{Linear, Tansig }	Discret
Larning rate (Lr)	[0.01 0.1]	Continue
First-order momentum (FOM)	[0.9 1]	Continue
Second-order momentum (SOM)	[0.9 1]	Continue
Batch size (BS)	{1,2...200}	Discret

*See table (1)

IV.7. Results and Disussion

IV.7. 1. Sidi Aich station

Table (IV.3) presents the optimized structures of ERNN, LSTM, GRU, and FFNN models based on PSO algorithm for Sidi Aich station. It can be seen from table (IV.3) that the applying hyperparameters of optimized models were provided for artificial neural networks models, respectively. GRU model with two-day-lag was selected as an optimal model for streamflow forecasting, and the hyperparameters can be chosen as follows: number of hidden layer = 1, number of neurons in hidden layer = 66, transfer function type in hidden layer = rectified linear unit (ReLU), transfer function type in output layer = linear (Linear), learning rate = 0.0312, first-order momentum = 0.8486, second-order momentum = 0.8139, and batch size = 46, respectively.

Table. IV.3. The optimal structure for FFNN, ELM, LSTM and GRU models using PSO for Sidi Aich station.

Hyper parameter	FFNN	ELM	LSTM	GRU
IM	Model I	Model IV	Model II	Model II
NHL	2	1	1	1
NNHL	9	8	61	66
AFHL	ReLU	ReLU	ReLU	ReLU
AFOL	Linear	Linear	Linear	Linear
Lr	0.0229	0.0116	0.0425	0.0312
FOM	0.9246	0.8439	0.8881	0.8486
SOM	0.9307	0.8564	0.8689	0.8139
BS	59	14	194	46

The results of RMSE, SNR, and NSE indices for different artificial neural networks models at Sidi Aich station are provided in table (IV.5). It can be found that GRU model with two-day-lag (RMSE = 35.620 m³ /sec, SNR = 0.6493, and NSE = 0.5783 for training phase and RMSE = 35.241 m³ /sec, SNR = 0.5159, and NSE = 0.7337 for testing phase) was better than other optimized models (i.e., ERNN model with four-day-lag, LSTM model with two-day-lag, and FFNN model with one-day-lag) during training and testing phases. On the other hand, the machine learning model (i.e., FFNN model with one-day-lag) provided the worst results (RMSE = 36.778 m³ /sec, SNR = 0.6706, and NSE = 0.5502 for training phase and RMSE = 37.440 m³ /sec, SNR = 0.5481, and NSE = 0.6994 for testing phase). Figure (IV.8. a-d) show the hydrographs and scatter diagrams between observed and forecasted streamflow using the optimized models at Sidi Aich station. It can be seen from figure (IV.8. a-d) that GRU model with two-day-lag yielded the best results compared to ERNN model with four-day-lag, LSTM model with two-day-lag, and FFNN model with one-day-lag, whereas FFNN model with one-day-lag showed the unexpected visual presentation for forecasting streamflow.

Table. IV.5. Comparison between the performance results obtained by the: FFNN, ELM, LSTM and GRU models in the training and testing phases for Sidi Aich and Ponteba Defluent stations.

		Traning phase			Testing phase		
		RMSE	SNR	NASH	RMSE	SNR	NASH
		(m3/s)			(m3/s)		
Sidi Aich	FFNN	36.778	0.6706	0.5502	37.440	0.5481	0.6994
	ELM	36.069	0.6573	0.5678	35.558	0.5206	0.7289
	LSTM	36.083	0.6577	0.5673	36.181	0.5297	0.7193
	GRU	35.620	0.6493	0.5783	35.241	0.5159	0.7337
Ponteba Defluent	FFNN	8.0989	0.5429	0.7052	11.976	0.3898	0.8480
	ELM	8.141	0.5457	0.7021	11.964	0.3894	0.8483
	LSTM	8.171	0.5477	0.6999	12.072	0.3929	0.8456
	GRU	7.796	0.5226	0.7268	11.074	0.3600	0.8703

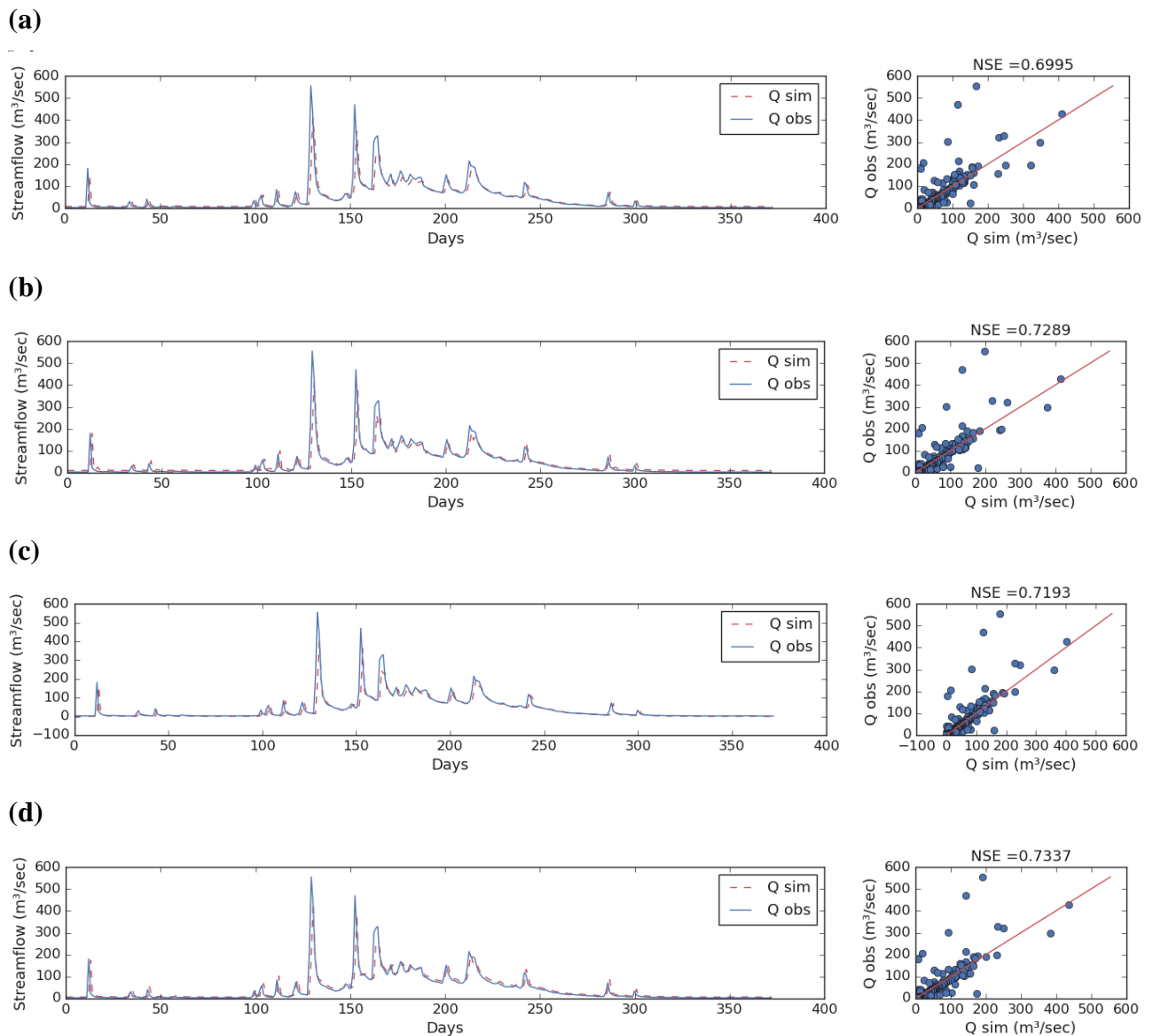


Figure. IV.8. Observed and simulated hydrographs and scatter plots during testing phase for Sidi Aich station : a) FFNN model, b) RELM model, c) ANFIS model, d) GRU model.

IV.7. 2. Ponteba Defluent station

Table (IV.4) shows the optimized structures of FRNN, LSTM, GRU, and FFNN models based on PSO algorithm for Ponteba Defluent station. It can be found from table (IV.4) that the hiring parameters of optimized models were suggested for artificial neural networks models, respectively. GRU model with two-day-lag was selected as an optimized model, and the

hyperparameters for GRU model with two-day-lag can be written as follows: number of hidden layer = 2, number of neurons in hidden layer = 196, transfer function type in hidden layer = sigmoid (Sigmoid), transfer function type in output layer = linear (Linear), learning rate = 0.0204, first-order momentum = 0.8269, second-order momentum = 0.8329, and batch size = 106, respectively. The results of RMSE, SNR, and NSE indices for different artificial neural networks models at Ponteba Defluent station are suggested in table (IV.5).

Table. IV.4. The optimal structure for FFNN, ELM, LSTM and GRU models using PSO for Ponteba Defluent station.

Hyper parameter	FFNN	ELM	LSTM	GRU
IM	Model I	Model I	Model I	Model II
NHL	1	1	2	2
NNHL	6	3	174	196
AFHL	ReLU	ReLU	ReLU	Sigmoide
AFOL	Linear	Linear	Linear	Linear
Lr	0,0186	0,0147	0,0691	0,0204
FOM	0,9220	0,8076	0,9130	0,8269
SOM	0,9033	0,8067	0,8184	0,8329
BS	36	55	194	106

It can be judged that GRU model with two-day-lag (RMSE = 7.796 m³ /sec, SNR = 0.5226, and NSE = 0.7268 for training phase and RMSE = 11.074 m³ /sec, SNR = 0.3600, and NSE = 0.8703 for testing phase) was better than other optimized models (i.e., ERNN model with one-day-lag, LSTM model with one-day-lag, and FFNN model with one-day-lag) during training and testing phases, whereas LSTM model with one-day-lag yielded the worst results (RMSE = 8.171 m³ /sec, SNR = 0.5477, and NSE = 0.6999 for training phase and RMSE = 12.072 m³ /sec, SNR = 0.3929, and NSE = 0.8456 for testing phase). Table 5 expresses that results of GRU model, were more accurate and efficient than ERNN, LSTM, and FFNN models for lag-

times streamflow forecasting. Figure (IV.9. a-d) show the hydrographs and scatter diagrams between observed and forecasted streamflow using the optimized models at Ponteba Defluent station. It can be found from figure (IV.9. a-d) that GRU model with twoday-lag provided the best results compared to ERNN model with one-day-lag, LSTM model with one-day-lag, and FFNN model with one-day-lag. While the forecasted streamflow using FFNN model with one-day-lag provided the worst accuracy.

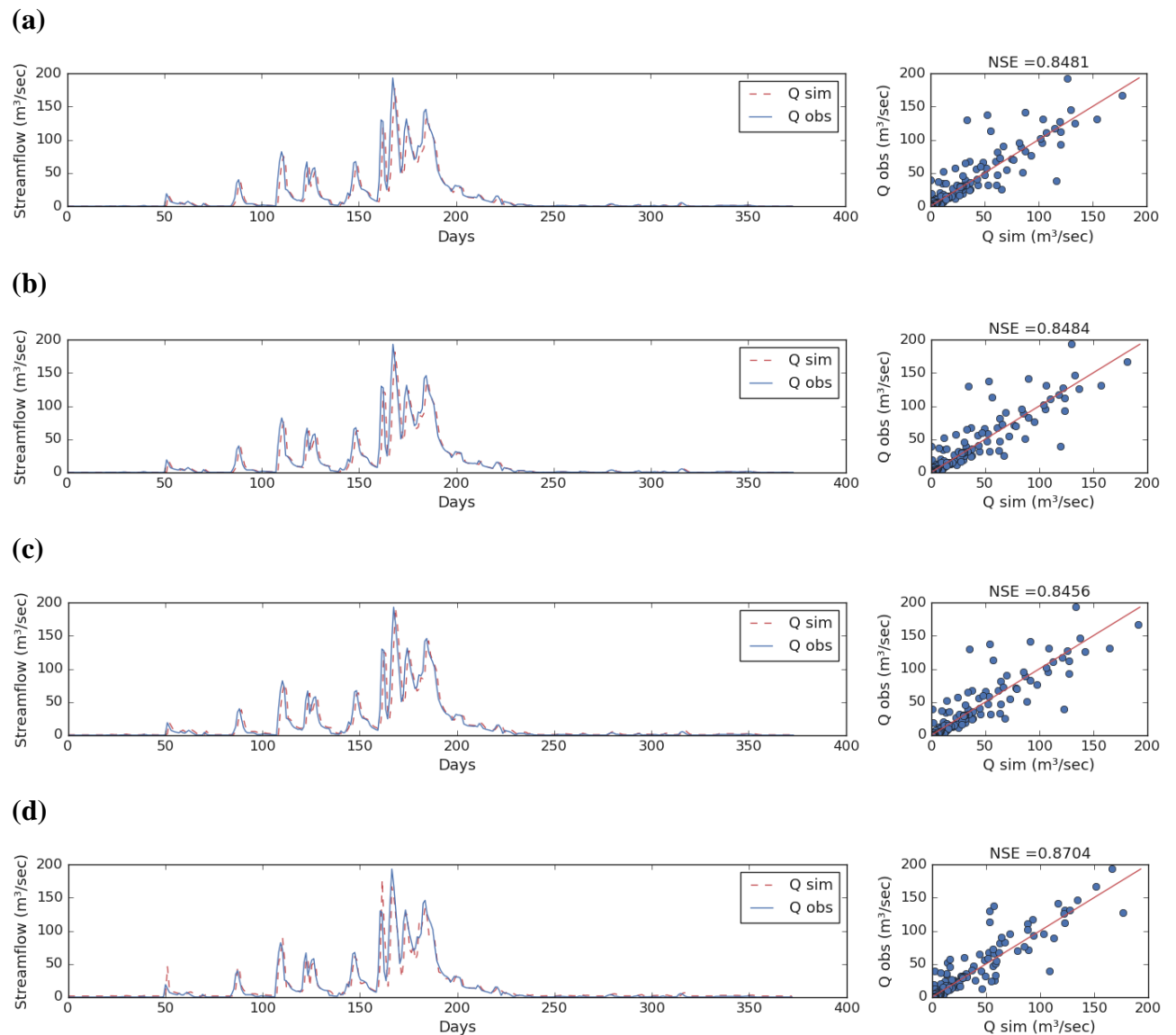


Figure. IV.9. Observed and simulated hydrographs and scatter plots during testing phase for Ponteba Defluent station : a) FFNN model, b) ELM model, c) LSTM model, d) GRU model.

IV.8. Conclusion

This research investigated the accuracy of artificial neural networks (i.e., FFNN, ERNN, LSTM, and GRU) models. To accomplish this research, daily streamflow data are collected from Sidi Aich and Ponteba Defluent stations, Algeria. The training and testing dataset were 83% and 17% of whole data (six years). Also, three statistical indices are utilized to compare the performance of artificial neural networks models for the different lag-times (i.e., one-,

two-, three-, and four-day-lag) streamflow forecasting. The model hyperparameters are optimized using particle swarm optimization (PSO) algorithm. Also, adaptive moment estimation (ADAM) is applied for model training.

The addressed models were assessed and evaluated by root mean square error (RMSE), signal-to-noise ratio (SNR), and Nash-Sutcliffe efficiency (NSE) statistical indices. Evaluating all models explained that the GRU model (RMSE = 35.241 m³ /sec, SNR = 0.5159, and NSE = 0.7337 at Sidi Aich station and RMSE = 11.074 m³ /sec, SNR = 0.3600, and NSE = 0.8703 at Ponteba Defluent station) was found to produce the accurate results compared to ERNN, LSTM, and FFNN models during testing phase for forecasting streamflow at both stations.

Chapter. V

Evolutionary Neuro-Wavelet and Neuro-Fuzzy systems for multi step ahead forecasting : case study in the Seybous River.

V.1. Introduction

Streamflow forecasting based on accurate measurements can be used to design flood mitigation structures for agricultural and urban basins, and build water allocation systems for agricultural, industrial, and commercial purposes (RezaieBalf et al., 2019; Samsudin et al., 2011). The complex and significant variability of streamflow can be explained using spatial and temporal characteristics. It has guided the evolution and application of different approaches for estimation, modeling, forecasting, and prediction (Martins et al., 2011). Forecasting of hydrological time series (e.g., streamflow, water stage, evaporation, and groundwater stage etc.) is important for understanding the internal relationship of natural processes (Krishna et al., 2011). Since the complexity of hydrological time series requires specific tools of nonlinear and non-stationary dynamic systems, the diverse forecasting methods have been proposed for hydrological forecasting (Sivakumar et al., 2001).

Data-driven approaches, although credible for hydrological forecasting, don't have the capability to depict physical processes, because they only consider an adequate selection of hydrological variables with temporal and input-output modification. Therefore, these approaches can be categorized as two types (i.e., classical and computational intelligence approaches) typically. The classical approaches can be exemplified as linear regression (LR), auto regressive integrated moving average (ARIMA), and ARIMA with exogenous input (ARIMAX) etc. The machine learning approaches can be classified as artificial neural networks (ANN), adaptive neurofuzzy inference systems (ANFIS), genetic programming (GP), gene expression programming (GEP), model tree (MT), extreme learning machines (ELM), support vector machines (SVM), and multivariate adaptive regression spline (MARS) etc. The field of machine learning approaches has undergone innovative changes for novel techniques of data simulation and processing (Chandwani et al., 2015).

For three decades, the ANN model based on neuron systems has been used for different types of hydrological forecasting. Various approaches have been applied to validate the accuracy and effectiveness of the ANN model for streamflow forecasting (Biswas and Jayawardena, 2014; Badrzadeh et al., 2013; Moradkhani et al., 2004; Cigizoglu, 2003; Abrahart and See, 2000). The ANFIS model (Jang, 1993), which has the merits of the ANN model and the fuzzy system, has been employed for streamflow forecasting (Talei et al., 2010; Nasr and Bruen, 2008; Keskin et al., 2006; Lohani et al., 2006).

Methodologies combining wavelet and machine learning approaches have been utilized for streamflow forecasting. Combined approaches have outstripped conventional models (Nourani et al., 2014). The wavelet-based machine learning approaches, including wavelet-based feed forward neural networks (WFFNN), wavelet-based support vector regression (WSVR), and wavelet-based adaptive neuro fuzzy inference system (WANFIS), have been effectively implemented for hydrological forecasting, including streamflow, water stage, runoff, and groundwater etc. (Seo et al., 2015; Kamruzzaman et al., 2013; Partal, 2009; Partal and Kisi, 2007). Many researchers have attempted to forecast streamflow using wavelet-based machine learning approaches (Zakhrouf et al., 2016, 2020; Shoaib et al., 2014; Badrzadeh et al., 2013; Nourani et al., 2013; Guo et al., 2011; Tiwari and Chatterjee, 2010).

The optimal structure of wavelet-based machine learning approaches can be constructed as a search method. A method for designing machine learning approaches using evolutionary optimization algorithm is proposed to format the best models. Evolutionary machine learning approaches for modeling hydrologic systems have been suggested by Zakhrouf et al. (2018), Kalteh (2015), Sahay and Srivastava (2014), Asadi et al. (2013).

K-fold cross validation (CV) method is one of the methods to assess the algorithmic generalization. The out-of-sample cross validation (OOS-CV) method is a frequently used

method for hydrological modeling. This paper attempts to develop a new approach combining wavelet transformation, machine learning approach, evolutionary optimization algorithm, and k-fold cross validation method for multi-step (days) (i.e., t+1, t+2, and t+3 days) streamflow forecasting in the Seybous River, North Algeria. The paper is divided into five chapters. The first part provides a brief introduction. The second part discusses data and application tools, including FFNN, ANFIS, WFFNN, GA, and k-fold CV, respectively. The third part applies the methodology, and results and discussion are presented in the fourth part. Conclusions are stated in the concluding part. This chapter includes partial contributions from the paper (Zakhrouf et al, 2020).

V.2. Materials and Methods

The aims of this chapter is to forecasting the streamflow of Seybous river flow using : Feedforward neural networks (FFNN), Adaptive neuro-fuzzy inference system (ANFIS), Wavelet-based neural networks (WFFNN) (see chapter I) and Genetic algorithm (GA) (see chapter III).

V.2.1. K-fold cross validation (CV)

The CV is a statistical methodology for comparing and evaluating training algorithms by separating data into two parts. One is utilized to train a model and the other is utilized to test it (Stone, 1974). K-fold CV assesses the generalization of algorithms in the evolutionary machine learning approach. Based on the category of k-fold CV, the data is first separated into k equally measured parts. Subsequently, k iterations of training and testing phases are performed such that a different part of the data is held-out for testing phase within each iteration, while the remaining $(k-1)$ parts are used for training phase (Zhao et al., 2018) Suppose we have a model with one or more unknown parameters $f(\alpha)$, and a data

set \hat{y} to which the model can be fitted. The primary method to estimate the tuning parameter α using k-fold CV divides the data into rougher parts (Fig. V.1). Since the model computes the mean squared error (MSE) for each $i= 1, 2, \dots, k$, the cross validation error (CVE) can be calculated as Eq. (V.1).

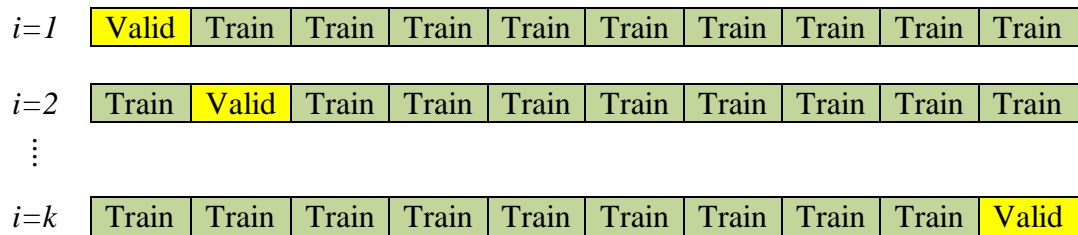


Figure. V.1. Representation of k-fold cross validation method

V.3. Study area and data description

The data for training and testing phases of the developed models were obtained from the Seybous River basin, Algeria. The basin is positioned between 36.0 N and 38.0 N (latitudes), and between 7.0 E and 8.0 E (longitudes), Algeria, North Africa, and covers a total catchment area = 6862,39 km² and Perimeter = 509.24 km and had a compactness index = 1,72. Seybous watershed drained by the Seybous river and which flows into the Mediterranean. The study area is characterized by a Mediterranean climate with hot and dry summer, and cold and rainy winter (Fig. V.1).

Daily streamflow data for 14 years (September, 1982 - August, 1996) were obtained from National Agency of Water Resources of Mirebeck (14 06 01) gauging station and were utilized for multi-step (days) streamflow forecasting. These selected multi-step (days) can be adequate, considering the rapid surface streamflow in the Mediterranean basin and the watershed size. For this purpose, the first ten years (70% of data) were utilized for the training

phase, and the second four years (30% of data) were utilized for the testing phase as shown in Fig (V.2).

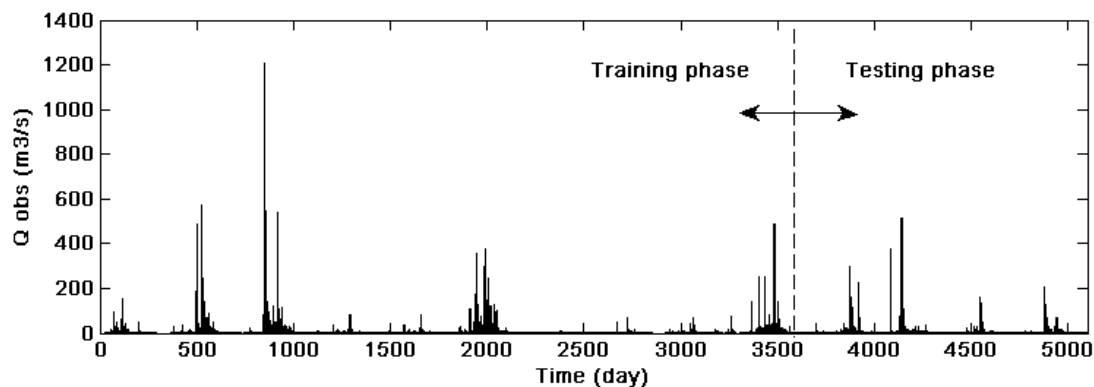


Figure. V.2. Observed streamflow hydrograph (14years)

V.4. Methodology

This study focuses on optimizing the structure of three machine learning approaches (i.e., FFNN, ANFIS, and WFFNN) using the evolutionary optimization algorithm (i.e., GA). The first step is to choose the GA chromosome. Each individual in the population represents a possible configuration for architecture of the machine learning approach. Based on the concept of evolutionary optimization theory, GA starts with a population of chromosomes, which evolve towards optimum solutions by GA operators, including selection, crossover, and mutation. These steps are reiterated from one generation to the next with the aim of arriving at the general optimal solution (Kim and Kim, 2018).

V.4.1 Models development

The machine learning approaches (i.e., FFNN, ANFIS, and WFFNN) were utilized for multi-step (days) streamflow forecasting using previous time series values. The training phase of machine learning approaches provides a non-linear matching between inputs and outputs, and is useful in identifying patterns utilizing complicated data (Liu and Chung, 2014). Since the

backpropagation (BP) algorithm cannot be guaranteed to find the minimum error margin, the convergence cannot occur with fast track. The solution for this problem can be applied with a fitness function that tests how well an architecture learns from the data. The fitness function can be expressed as Eq. (V.2).

$$F = \text{Min}[CVE] \quad (\text{V.2})$$

where *CVE* is the cross validation error given by the equation (V.1) and *F* denotes the fitness function.

In general, GA can significantly reduce the weakness of BP algorithm. The data utilized for the training phase (the 75% of the data) were sub-divided into 10 subsets (10-fold cross validation). Real coding was utilized to find the favorable topology for the FFNN, ANFIS, and WFFNN models. The coding to find the best architecture and parameters of machine learning approaches (i.e., FFNN, ANFIS, and WFFNN) is described as follows.

V.4.2 Coding for the FFNN, ANFIS, and WFFNN models

Feedforward neural networks (FFNN) models with two hidden layers and one neuron in final layer were utilized. A chromosome was built from a series of genes (Fig. V.3) to find: the input delay (*D*), the number of neuron in the hidden layers (*NHL1* and *NHL2*), the activation functions in the hidden and final layers (*AFHL1*, *AFHL2*, and *AFOL*) including linear transfer function (*Purelin*), symmetric saturating linear transfer function (*Satlins*), log-sigmoid transfer function (*Logsig*), and hyperbolic tangent sigmoid transfer function (*Tansig*), respectively, and the initial connection coefficients of weights and bias (*IWB*).

<i>Genes</i>	<i>Description</i>	<i>Values</i>
g_1	D	{1,2, ..., 5}
g_2	NHL1	{2,3, ..., 10}
g_3	NHL2	{2,3, ..., 10}
g_4	AFHL1	{Satlins, Purelin, Logsig, Tansig}
g_5	AFHL2	{Satlins, Purelin, Logsig, Tansig}
g_6	AFOL	{ Satlins, Purelin, Logsig, Tansig}
g_7	IWB	[-1 1]
g_8		[-1 1]
\vdots		\vdots
g_n		[-1 1]

Figure. V.3. Chromosome encoding for FFNNs model

A chromosome of ANFIS model using a series of genes was created to find different parameters (Fig. V.4): the input delay (D); the number of membership functions (NMF); the type of membership functions (TMF) including Π -shaped (Pimf), Trapezoidal-shaped (Trapmf), Triangular-shaped (Trimf), Gaussian curve (Gaussmf), and Built-in Gaussian function (Dsigmf); the definition of if-then (AND operation/OR operation) rules type (DRT); and the firing strength of a rule (FSR). In this study, two methods were used as firing strength of AND rule, including Minimum (Min) and Product (Prod). Also, two methods were used as firing strength OR rule, including Maximum (Max) and the probabilisticOr method (Probor). For the membership functions type, there are many kinds of membership functions (MFs).

<i>Genes</i>	<i>Description</i>	<i>Values</i>
g_1	D	{1,2, ..., 5}
g_2	NMF	{2, 3}
g_3	TMF	{Pimf, Trapmf, Trimf, Gaussmf, Dsigmf}
g_4	FSR	{(Prod, Max) ; (Prod, Probor) ; (Min, Max) ; (Min, Probor)}
g_5	DRT	{And, Or}

Figure.V.4. Chromosome encoding for ANFIS model

A chromosome of different parameters for wavelet-based neural networks (WFFNN) model was created from a series of genes (Fig. V.5) to find: the input delay (D); the type of mother wavelet (TMW) based on the five most frequently used wavelet families, including Haar (Har), Daubechies (Db), Coiflets (Coif), Symlets (Sym), and Biorthogonal (Bior); the number of neuron in hidden layers (NHL1 and NHL2); the activation functions in hidden and output layers (AFHL1, AFHL2, and AFOL), including linear transfer function (Purelin), symmetric saturating linear transfer function (Satlins), logsigmoid transfer function (Logsig) and hyperbolic tangent sigmoid transfer function (Tansig); and the initial connection weights and bias coefficients (IWB).

<i>Genes</i>	<i>Description</i>	<i>Values</i>
g_1	D	{1,2, ..., 5}
g_2	MWT	{Har, Db(4...9), Sym(3... 8), Coif(2... 5), Bior(1.3, 1.5, 2.2)}
g_3	NHL1	{2, 3, ..., 10}
g_4	NHL2	{2, 3, ..., 10}
g_5	AFHL1	{Satlins, Purelin, Logsig, Tansig}
g_6	AFHL2	{Satlins, Purelin, Logsig, Tansig}
g_7	AFOL	{Satlins, Purelin, Logsig, Tansig}
g_8		[0 1]
g_9		[0 1]
\vdots	IWB	\vdots
g_n		[0 1]

Figure. V.5. Chromosome encoding for WFFNNs model

V.5. Measures of accuracy

To assess the performance of three different machine learning approaches (i.e., FFNN, ANFIS, and WFFNN) to forecast multi-step (days) streamflow during the testing phase, four statistical indices (measures of accuracy) were applied, including root mean square error (RMSE), Nash-Sutcliffe efficiency (NSE), correlation coefficient (R), and peak flow criteria (PFC).

RMSE would vary from zero to a large value which presents perfect forecasting by the difference between observed and forecasted streamflow. NSE is considered for evaluating the ability of hydrological models (Rezaie-Balf et al., 2019; Nash and Sutcliffe, 1970). R is an assessment of the precision of hydrologic modeling and is used for comparisons of alternative models. A perfect matching produces $R = 1.0$ (Kim and Kim, 2008). These statistical indices (i.e., RMSE, NSE, and R) may not provide the model performance for extreme streamflow events (e.g., floods and drought). Therefore, it is fundamental to evaluate the extreme

distributions using PFC for forecasting extreme events (Rezaie-Balf et al., 2019). PFC plays an important role in following the extreme events to achieve the efficient model. The RMSE, NSE, R, and PFC indices can be represented as Eqs. (V.3- V.6).

$$RMSE = \sqrt{\frac{\sum_{i=1}^N (Qt_i - \hat{Q}t_i)^2}{N}} \quad (V.3)$$

$$NSE = \left(1 - \frac{\sum_{i=1}^N (Qt_i - \hat{Q}t_i)^2}{\sum_{i=1}^N (Qt_i - \bar{Q}t)^2}\right) \quad (V.4)$$

$$R = \frac{\sum_{i=1}^N (Qt_i - \bar{Q}t)(\hat{Q}t_i - \bar{Q}t)}{\sqrt{\sum_{i=1}^N (Qt_i - \bar{Q}t)^2 \sum_{i=1}^N (\hat{Q}t_i - \bar{Q}t)^2}} \quad (V.5)$$

$$PFC = \frac{(\sum_{i=1}^{Tp} (Qt_i - \hat{Q}t_i)^2 \cdot Qt_i^2)^{1/4}}{(\sum_{i=1}^{Tp} Qt_i^2)^{1/2}} \quad (V.6)$$

where Qt_i is the measured flow rate value, $\hat{Q}t_i$ is the flow rate calculated by the model, $\bar{Q}t$ is the average flow measured, $\tilde{Q}t$ is the average flow simulated and N is the number of data, and Tp is the number of peak stream flows greater than one third of the observed mean peak flow.

V.6. Results and Discussion

Table (V.1) presents the optimal structure of FFNN model using GA which can be found from table (V.1) for (t+1) day forecasting as follows; input delay = 3, number of neurons (1st hidden layer) = 9, number of neurons (2nd hidden layer) = 8, activation function (1st hidden layer) = log-sigmoid transfer function, activation function (2nd hidden layer) = linear transfer function, and activation function (final layer) = linear transfer function. For (t+2) day forecasting, the following can be suggested; input delay = 3, number of neurons (1st hidden layer) = 9, number of neurons (2nd hidden layer) = 8, activation function (1st hidden layer) = log-sigmoid transfer function, activation function (2nd hidden layer) = log-sigmoid transfer function, and activation

function (final layer) = linear transfer function. For (t+3) day forecasting, the following can be provided; input delay = 5, number of neurons (1st hidden layer) = 5, number of neurons (2nd hidden layer) = 8, activation function (1st hidden layer) = symmetric saturating linear transfer function, activation function (2nd hidden layer) = log-sigmoid transfer function, and activation function (final layer) = linear transfer function.

Table. V.1. The optimal structure for FFNNs model using GA.

FFNNs parameters	t+1	t+2	t+3
D	3	3	5
NHL1	9	9	5
NHL2	8	8	8
AFHL1	Logsig	Logsig	Satlin
AFHL2	Purelin	Logsig	Logsig
AFOL	Purelin	Purelin	Purelin

Table (V.2) shows the optimal structure of ANFIS model using GA which can be expressed from table (V.2) for (t+1) day forecasting as follows; input delay = 1, number of membership functions = 3, type of membership functions = Gaussian curve, firing strength of a rule = product, and definition of if-then rules type = and. For (t+2) day forecasting, the following can be suggested; input delay = 2, number of membership functions = 3, type of membership functions = trapezoidal-shaped, firing strength of a rule = product, and definition of if-then rules type = and. For (t+3) day forecasting, the following can be provided; input delay = 2, number of membership functions = 2, type of membership functions = Gaussian curve, firing strength of a rule = probabilisticOr, and definition of if-then rules type = or.

Table. V.2. The optimal structure for ANFIS model using GA

ANFIS parameters	t+1	t+2	t+3
D	1	2	2
NMF	3	3	2
TMF	Gaussmf	Trapmf	Gaussmf
FSR	Prod	Prod	Probor
DRT	And	And	Or

Table (V.3) proposes the optimal structure of WFFNN model using GA which can be supposed from table (V.3) for (t+1) day forecasting as follows; input delay = 4, type of mother wavelet = Symlets 7, number of neurons (1st hidden layer) = 5, number of neurons (2nd hidden layer) = 10, activation function (1st hidden layer) = log-sigmoid transfer function, activation function (2nd hidden layer) = log-sigmoid transfer function, and activation function (final layer) = linear transfer function. For (t+2) day forecasting, the following can be suggested; input delay = 4, type of mother wavelet = Daubechies 9, number of neurons (1st hidden layer) = 8, number of neurons (2nd hidden layer) = 10, activation function (1st hidden layer) = symmetric saturating linear transfer function, activation function (2nd hidden layer) = log-sigmoid transfer function, and activation function (final layer) = linear transfer function. For (t+3) day forecasting, the following can be provided; input delay = 5, type of mother wavelet = Symlets 6, number of neurons (1st hidden layer) = 8, number of neurons (2nd hidden layer) = 7, activation function (1st hidden layer) = linear transfer function, activation function (2nd hidden layer) = symmetric saturating linear transfer function, and activation function (final layer) = linear transfer function.

Table.V.3. The optimal structure for WNNs model using GA

WNNs parameters	t+1	t+2	t+3
D	4	4	5
MWT	sym7	db9	Sym6
NHL1	5	8	8
NHL2	10	10	7
AFHL1	Logsig	Satlin	Purelin
AFHL2	Logsig	Logsig	Satlin
AFOL	Purelin	Purelin	Purelin

Table (V.4) suggests the performances of FFNN, ANFIS, and WFFNN models for different multi-step (days) (i.e., t+1, t+2, and t+3 days) forecasting. The statistical results of standalone (i.e., FFNN and ANFIS) models yielded similar performances based on RMSE, NSE, R, and PFC indices for training and testing phases. The performances of standalone models were not better than those of hybrid model (i.e., WFFNN). For example, the values of RMSE and PFC for the WFFNN model (RMSE = 8.590 m³ /sec and PFC = 0.252, (t+1) day forecasting) were lower than those of FFNN (RMSE = 19.120 m³ /sec and PFC = 0.446, (t+1) day forecasting) and ANFIS (RMSE = 18.520 m³ /sec and PFC = 0.444, (t+1) day forecasting) models in the testing phase. In addition, the values of NSE and R for the WFFNN model (NSE = 92.000% and R = 0.969, (t+1) day forecasting) were higher than those of FFNN (NSE = 60.400% and R = 0.783, (t+1) day forecasting) and ANFIS (NSE = 62.860% and R = 0.793, (t+1) day forecasting) models in the testing phase. The performances of hybrid model were superior to those of standalone models. As the multi-step (days) for the three models increased from (t+1) to (t+3) days, the model accuracy decreased. The hybrid model applied sub-time series by using WT as model input, while the standalone models utilized the original time series as model input without WT. It can be suggested from table (V.4) that the hybrid model using

sub-time series as input data of the standalone models can improve the performance of conventional standalone models.

Table. V4. Comparison between the performance results obtained by the models: FFNNs, ANFIS and WFFNNs in the training and testing phases.

		Training				Testing			
		RMSE (m3/s)	NSE	R	PFC	RMSE (m3/s)	NSE	R	PFC
ANN	t+1	23,85	0,701	0,837	0,463	19,12	0,604	0,783	0,446
	t+2	30,99	0,495	0,704	0,513	25,39	0,302	0,560	0,501
	t+3	37,27	0,270	0,520	0,645	30,80	-2,71	0,352	0,552
ANFIS	t+1	25,61	0,655	0,810	0,479	18,52	0,628	0,793	0,444
	t+2	35,91	0,322	0,568	0,626	26,09	0,263	0,535	0,510
	t+3	38,47	0,222	0,472	0,653	27,14	0,202	0,457	0,546
WNN	t+1	3,53	0,993	0,997	0,103	8,59	0,9200	0,969	0,252
	t+2	5,23	0,985	0,993	0,128	13,36	0,807	0,917	0,318
	t+3	7,95	0,966	0,983	0,103	12,95	0,818	0,907	0,391

Fig (V.6) shows the scatter diagrams for the FFNN, ANFIS, and WFFNN models in the testing phase. Fig (V.7) presents the relative errors of peak flow for the FFNN, ANFIS, and WFFNN models in the testing phase. Figs (V.6) and (V.7) show that the performance of hybrid model was superior to that of standalone models.

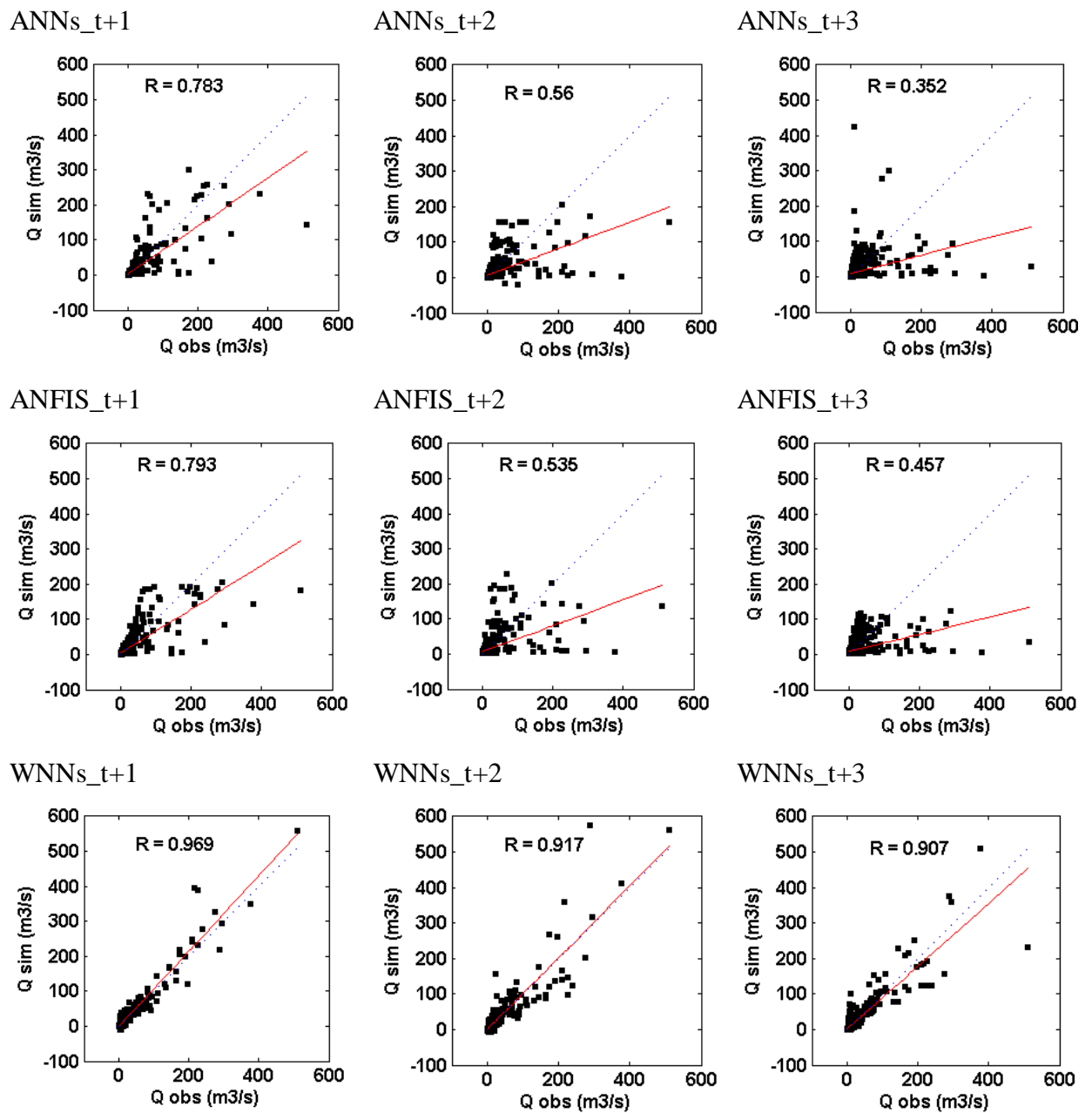


Figure. V.6. Scatter diagram for FFNNs, ANFIS and WFFNNs models (testing phase)

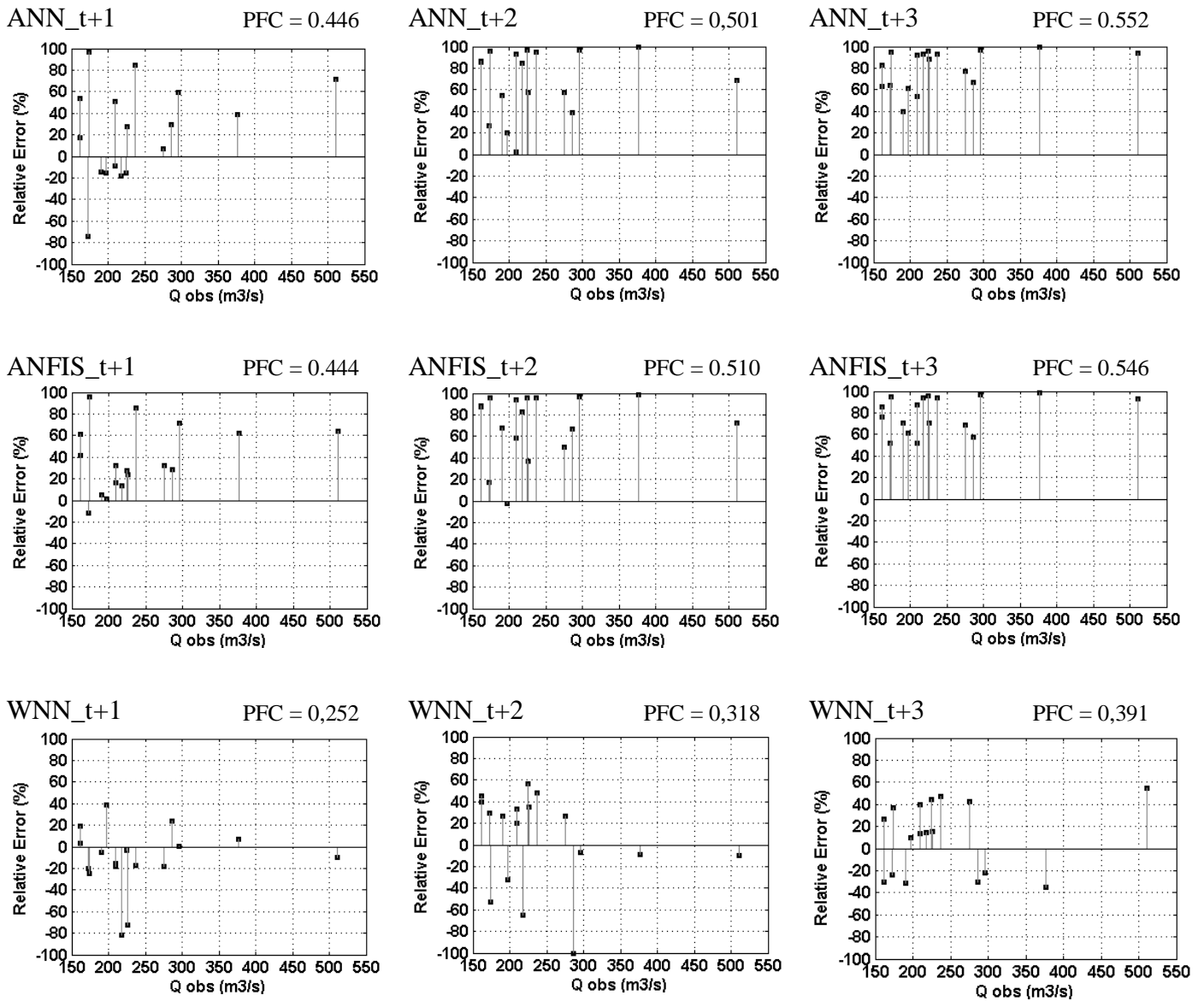


Figure. V.7. Relative errors of the peak flows for FFNNs, ANFIS and WFFNNs models (testing phase)

V.7. Conclusion

This study suggests three artificial neural networks approaches, including feed forward neural networks (FFNN), adaptive neurofuzzy inference system (ANFIS), and wavelet-based neural

networks (WFFNN) models for multi-step (days) streamflow forecasting in Seybous River basin, Algeria. The accuracy and effectiveness of developed models (i.e., FFNN, ANFIS, and WFFNN) are evaluated based on four statistical indices, including root mean square error (RMSE), Nash-Sutcliffe efficiency (NSE), correlation coefficient (R), and peak flow criteria (PFC).

Based on the combination of evolutionary optimization algorithm and k-fold cross validation, the statistical results of hybrid (i.e., WFFNN) model are superior to those of standalone (i.e., FFNN and ANFIS) models for different multi-step (days).

The values of RMSE and PFC for the WFFNN model (e.g., RMSE = 8.590 m³ /sec, PFC = 0.252 for (t+1) day, testing phase) were lower than those of FFNN (e.g., RMSE = 19.120 m³ /sec, PFC = 0.446 for (t+1) day, testing phase) and ANFIS (e.g., RMSE = 18.520 m³ /sec, PFC = 0.444 for (t+1) day, testing phase) models, while the values of NSE and R for WFFNN model were higher than those of FFNNs and ANFIS models.

Also, the results of (t+1) day streamflow forecasting are superior to those of (t+2) and (t+3) days, based on statistical indices and scatter diagrams. The combination of machine learning approach and data pre-processing technique based on the evolutionary optimization algorithm and k-fold cross validation can be a potential implement for accurate multistep (days) streamflow forecasting. Hybrid methodologies combining diverse machine learning approaches and data preprocessing technique based on other evolutionary optimization algorithms and cross validations can be recommended for further studies.

Chapter. VI

**Applied Evolutionary Neuro-Wavelet by three
diffrents strategies for multi step ahead forecasting.
Case study : Chellif River.**

V.1. Introduction

Accurate estimates of streamflow can be utilized in several water engineering problems such as designing food protection works for urban areas and agricultural land, and optimizing the water allocation for different purposes including agriculture, municipalities, and hydropower generation (Rezaie-Balf et al. 2019; Seo et al. 2018; Samsudin et al. 2011). The complexity of natural processes and the lack of data available for modeling streamflow require the utilization of specific implements for nonlinear and non-stationary natural phenomenon (Chu et al. 2016; Shoaib et al. 2015). Time series forecasting is one of the most and important methodologies utilized in streamflow modeling (Rezaie-Balf and Kisi 2017a; Seo et al. 2015; Krishna et al. 2011). The problem complexity rises when the models are applied for days/months forecasting in advance. Streamflow forecasting utilizing available multi-time-ahead series is a common task (Ghorbani et al. 2018; Karimi et al. 2018; Wang et al. 2009).

In recent years, the field of computational intelligence has promoted revolutionary changes to forecast the streamflow of complex and non-stationary time series in the development of non-conventional techniques (Delafrouz et al. 2018; Yu et al. 2018; Shafaei and Kisi 2016). In addition, various data pre-processing techniques have been utilized to enhance the accuracy of streamflow forecasting. The underlying techniques include principal component analysis (PCA) (Ravikumar and Somashekar 2017; Hu et al. 2007), continuous wavelet transform (CWT) (Deo et al. 2017; Rezaie-Balf et al. 2017; Sang et al. 2013), moving average (MA) (Yuan et al. 2017), wavelet multi-resolution analysis (WMRA) (Zakhrouf et al. 2018), maximum entropy spectral analysis (MESA) (Benedetto et al. 2015) and singular spectrum analysis (SSA) (Wu and Huang 2009; Baydaroglu et al. 2018). Recently, the combination of discrete wavelet transform (DWT) and artificial neural networks (ANNs) approaches has been

accomplished as the successful alternative for hydrological modeling and forecasting from the previous literatures (Abdollahi et al. 2017; Ravansalar et al. 2016, 2017; Zakhrouf et al. 2016).

In many engineering problems, since streamflow forecasting utilizing simple time series (e.g., one-day-ahead) may not provide enough information, the multi-day-ahead (e.g., 2, 3, and 4-day-ahead etc.) forecasting is required to understand the physical processes of streamflow time series clearly (Chang et al. 2007). For this purpose, this research is designed to investigate the efficiency and capability of the hybrid model to forecast the multi-day-ahead streamflow. The wavelet-based feedforward neural networks (WFFNNs) model optimized by genetic algorithm (GA) is developed to forecast the multi-day-ahead streamflow in the Chellif River, Algeria. Also, this study applies the reconstruction of three evolutionary strategies [i.e., multi-input multi-output (MIMO), multi-input single-output (MISO), and multi-input several multi-output (MISMO)] utilizing the WFFNNs-GA model. The contributions of this research can be classified as two parts. First, the combinational approaches (i.e., DWT, FFNNs, and GA) based on three evolutionary strategies (i.e., MIMO, MISO, and MISMO) are proposed, respectively. Second, the authors present the hybrid model for forecasting the multi-day-ahead streamflow of the Chellif River, Algeria. The performance of developed models is evaluated utilizing five statistical indices and diagnostic plots. Finally, the conclusion and future research are given.

This chapter includes partial contributions from the paper (Zakhrouf, et al 2020 b).

VI.2. Materials and Methods

In this chapter we proposed to forecast the streamflow of Ponteba Defluent gauging station the following models and methods : The Feedforward neural networks (FFNNs), the wavelet Feedforward neural networks (WFFNNs) models (see chapter I), the genetic algorithm (see chapter III) and the k-fold cross validation (see chapter V).

VI.3. Methodology description

For a given time series $\{ Q_1, Q_2, \dots, Q_t \}$ to perform H step ahead streamflow forecasting, independent method consists of estimating a set of H forecasting networks, each returning a direct forecasting of Q_{t+h} with $h \in \{1, \dots, H\}$. Joint approach replaces the H networks of direct approach with one multi-output networks $\{ Q_{t+1}, Q_{t+2}, \dots, Q_{t+H} \}$. MISMO strategy can be expressed as a combination between two approaches. The independent and joint approach can be indeed seen as two distinct instances of the same forecasting approach. In the independent case, the number of forecasting tasks is equal to the size of the horizon H , and the size of the outputs is 1. In the joint case, the number of forecasting tasks is equal to one, and the size of the output is H , respectively. Intermediate configurations can be considered by transforming the original task into $n = H/s$ forecasting tasks, each with multiple outputs of size s , where $s \in \{1, \dots, H\}$. MISMO trades of the property of preserving the stochastic dependency between future values with a greater flexibility of the predictor (Ben Taieb et al. 2010). In this study, for the multi-day-ahead streamflow forecasting utilizing the FFNNs-GA and WFFNNs-GA models, three evolutionary strategies, including multi-input multioutput (MIMO), multi-input single-output (MISO), and multi-input several multi-output (MISMO), were applied. In this category, the MISO strategy was employed utilizing independent method (Fig. VI.1.a), and the MIMO and MISMO strategies were suggested utilizing joint method (Fig. VI.1.b) (Kline 2004).

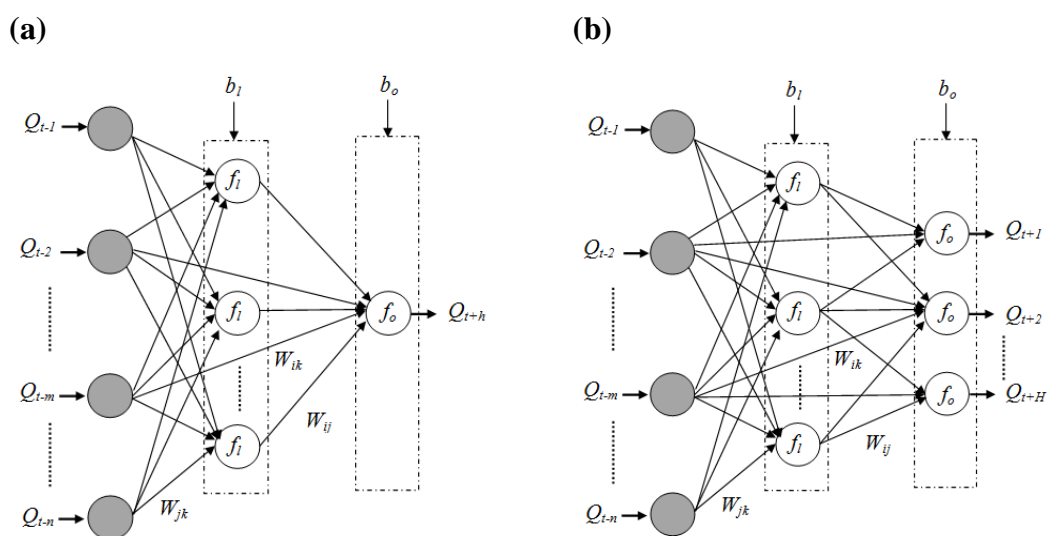


Figure. VI.1. Conventional FFNNs architecture (a) Independent method, (b) Joint method.

VI.4. FFNNs-GA and WFFNNs-GA models

Evolutionary algorithms [e.g., genetic algorithm (GA), evolutionary strategy (ES), and programming (EP)] are a class of stochastic searches and optimization techniques (Ding et al. 2013). This study aims to take advantage of GA properties including selection, crossover, and mutation to find the global minimum errors (Kim and Kim 2008; Jain and Srinivasulu 2004; Holland 1992; Goldberg and Deb 1991; Goldberg and Holland 1988). Therefore, a method for designing the FFNNs and WFFNNs models' structures utilizing GA was proposed to construct the best models. The GA and BP algorithms were applied to optimize the FFNNs and WFFNNs model. In addition, the ability of BP algorithm can find local minimum errors. For this purpose, the steps can be described as follows:

Step 1, Encoding

The different parameters of WFFNNs model are coded by a chromosome model developed from a series of genes as shown in Fig(VI.2). Each gene represents the input delay (D), type of

the mother wavelet (TMW), decomposition levels (L), number of neurons in the hidden layers (NHL1 and NHL2), activation functions in hidden and output layers (AFHL1, AFHL2, and AFOL), and the initial connection weights and bias coefficients (IWB), respectively. This study compared the effects of 20 selected wavelet functions from the most frequently utilized mother wavelets [e.g., Haar (Har), Daubechies (Db), Coifets (Coif), Symmlets (Sym), and Biorthogonal (Bior)]. For analysis of DWT method, Daubechies, Symmlets, and Coifets wavelets have been commonly utilized as mother wavelets in waveletbased hydrologic studies (Seo et al. 2015; Evrendilek 2014; Santos et al. 2014; Adamowski and Sun 2010; Tiwari and Chatterjee 2010). In addition, the effects of activation functions for the accuracy of WFFNNs-GA model were demonstrated utilizing the four activation functions, including linear transfer function (Purelin), symmetric saturating linear transfer function (Satlins), log-sigmoid transfer function (Logsig), and hyperbolic tangent sigmoid transfer function (Tansig), respectively.

<i>Genes</i>	<i>Description</i>	<i>Values</i>
g_1	D	{1,2, ..., 14}
g_2	MWT	{Har, Db(4...9), Sym(3... 8), Coif(2... 5), Bior(1.3, 1.5, 2.2)}
g_3	L	{1,2, ..., 6}
g_4	NHL1	{2, 3, ..., 10}
g_5	NHL2	{2, 3, ..., 10}
g_6	AFHL1	{Satlins, Purelin, Logsig, Tansig}
g_7	AFHL2	{Satlins, Purelin, Logsig, Tansig}
g_8	AFOL	{Satlins, Purelin, Logsig, Tansig}
g_9		[-1 1]
g_{10}		[-1 1]
\vdots	IWB	\vdots
g_n		[-1 1]

Figure. VI.2. Chromosome encoding for WFFNNs model

Step 2, Initializing random individual topology (chromosomes).

Step 3, Training each individual by BP algorithms utilizing k-fold cross-validation.

Step 4, Calculating the fitness values With regard to the fitness function, it is based on the mean squared error (MSE) over a training dataset. With k-fold cross-validation method, datasets are divided into k parts.

Note that each dataset should be divided into two parts of the calibration and validation phases. In each run, one fold of dataset is allocated for the validation phase, and k-1 folds are allocated for calibrating the model. This process is repeated k times. Real error of this model can be estimated by averaging the error of k runs of the model, which is represented by Eq. (VI.1)

$$E = \frac{1}{k} \sum_{i=1}^k MSE_i \quad (VI.1)$$

where MSE_i is the mean squared error for each $i= 1, 2, \dots k$. In the case of our study the number of folds is equal a 10 folds.

Step 5, Selecting the best individual mechanism.

Step 6, Crossover and mutation operators.

Step 7, Replacing the current population by the newly generated offsprings.

Step 8, Stopping criteria; otherwise, go to step 3.

VI.5. Study area and data

The description of study watershed is showed in the chapter IV.

The datasets (14 years) were collected from Ponteba Defuent station (01 22 03) of the National Agency of Water Resources. The first 10-year data (70% of the whole dataset) were employed for the model calibration, and the remaining 4 years (30% of the whole dataset)

were applied for the model validation in this study (Fig. VI.3). The size and rapid slope of the Chellif River basin can restrict the ranges of multi-time-ahead streamflow as 1-, 2-, 3-, and 4-day-ahead forecasting. Also, the authors applied three evolutionary strategies as MISO (four model), MISMO (two models), and MIMO (one model), respectively.

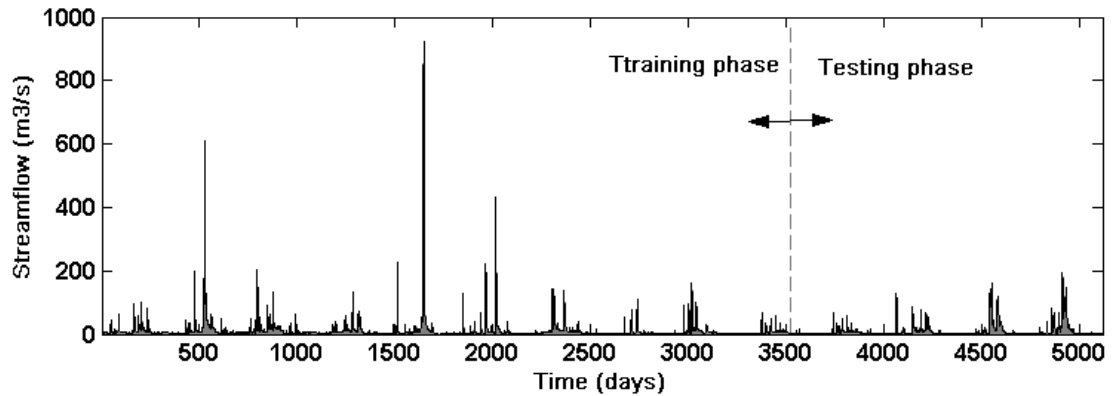


Figure. VI.3. Observed streamflow hydrograph (14years)

VI.6. Measures of accuracy

To assess the performance of the stand-alone and hybrid models for streamflow forecasting based on the different strategies during the calibration and validation phases, five statistical indices are presented in equations (VI.2-VI.6), including root mean squared error (RMSE), signal-to-noise ratio (SNR), correlation coefficient (R), Nash–Sutcliffe efficiency (NSE), and peak flow criteria (PFC). In forecasting models, the discrepancy between observed and forecasted values can be shown utilizing RMSE index. This metric would vary from zero to a large value which presents perfect forecasting by the difference between observed and forecasted values. NSE index is taken into account to evaluate the ability of hydrological models. The higher value of it demonstrates a perfect fit between observed and forecasted streamflow (Nash and Sutcliffe 1970).

SNR index can be defined as the ratio of meaningful information to the unwanted one. It enables the comparison of modeling uncertainty to the effects of hydrologic scenarios. If SNR index is equal to zero, it indicates the perfect performance between observed and forecasted streamflow. However, the higher value of it explains the unacceptable performance (Bormann 2005). R index is a measure of the accuracy of a hydrologic modeling and is utilized for comparisons of alternative models. A high R value addresses a good model performance, and vice-versa. A perfect match between the observed and forecasted streamflow yields $R=1.0$ (Kim and Kim 2008). NSE index can range from negative infinite ($-\infty$) to 100%. If NSE equal to 100%, it corresponds to a perfect fits of forecasted streamflow. If NSE index is equal to zero, it has no more accurate forecasting, and the model performance cannot be acceptable. If NSE can range from minus 100% to zero, it indicates the unacceptable performance. The addressed statistical indices (e.g., RMSE, SNR, R, and NSE) may not illustrate the models' performance in case of the extreme events (e.g., flood and drought) of streamflow. That is, one model may forecast the mean values of streamflow accurately, but cannot forecast the high and low values of streamflow. As a result, the statistical indices such as NSE, RMSE, Willmott's index of agreement (WI) (Willmott 1984), and Legates–McCabe's index (LMI) (Legates and McCabe 1999) cannot present an adequate diagram. Therefore, it is essential to assess and monitor the extreme values utilizing PFC index for forecasting the extreme events. PFC index plays a significant role in monitoring the extreme events including flood to achieve the efficient model. It is noticeable to say that PFC equal to zero represents a perfect fit of model.

$$RMSE = \sqrt{\sum_{i=1}^N (Qt_i - \hat{Q}t_i)^2 / N} \quad (VI.2)$$

$$NSR = \frac{\sqrt{\sum_{i=1}^N (Qt_i - \hat{Q}t_i)^2 / \gamma}}{\delta} \quad (VI.3)$$

$$R = \frac{\sum_{i=1}^N (Qt_i - \bar{Q}t)(\hat{Q}t_i - \bar{Q}t)}{\sqrt{\sum_{i=1}^N (Qt_i - \bar{Q}t)^2 \sum_{i=1}^N (\hat{Q}t_i - \bar{Q}t)^2}} \quad (\text{VI.4})$$

$$NSE = \left(1 - \frac{\sum_{i=1}^N (Qt_i - \hat{Q}t_i)^2}{\sum_{i=1}^N (Qt_i - \bar{Q}t)^2} \right) \quad (\text{VI.5})$$

$$PFC = \frac{(\sum_{i=1}^{Tp} (Qt_i - \hat{Q}t_i)^2 \cdot Qt_i^2)^{1/4}}{(\sum_{i=1}^{Tp} Qt_i^2)^{1/2}} \quad (\text{VI.6})$$

Where Qt_i is the measured flow rate value, $\hat{Q}t_i$ is the flow rate calculated by the model, $\bar{Q}t$ is the average flow measured, $\bar{Q}t$ is the average flow simulated and N is the number of data, γ is the number of degrees of freedom, and δ is the standard deviation of the observed flow, Tp is the number of peak stream flows greater than one third of the observed mean peak flow.

VI.7. Results and discussion

Table (VI.1) presents the optimal structure of WFFNNs-GA model based on the MISO strategy in this study. It can be found from Table (VI.1) that the variables of WFFNNs-GA model were demonstrated for the multi-day-ahead (e.g., 1-, 2-, 3-, and 4-day-ahead) forecasting, respectively. For example, the variables of WFFNNs-GA model with 1-day-ahead streamflow forecasting were determined as follows; input delay=12, the type of mother wavelet=Db9, decomposition level = 4, number of neuron in the first hidden layer=9, number of neuron in the second hidden layer=5, activation function in the first hidden layer=log-sigmoid transfer function, activation function in the second hidden layer=linear transfer function, and activation function in output layer = linear transfer function, respectively.

Table. VI.1. The optimal structure for WFFNNs model based MISO strategy using GA

Parametres	Model (t+1)	Model (t+2)	Model (t+3)	Model (t+4)
D	12	11	12	11
MWT	Db9	Coif5	Db6	Db9
L	4	5	4	5
NHL1	9	6	6	10
NHL2	5	5	4	8
AFHL1	Logsig	Tansig	Logsig	Tansig
AFHL2	Purelin	Logsig	Satlin	Logsig
AFOL	Purelin	Purelin	Purelin	Purelin

Table (VI.2) proposes the optimal structure of WFFNNs-GA model based on the MISMO strategy in this study. It can be seen from Table 3 that the variables of WFFNNs-GA model with 1- and 2-day-ahead streamflow forecasting were provided as follows: input delay=12, the type of mother wavelet= Db9, decomposition level=5, number of neurons in the first hidden layer=9, number of neurons in the second hidden layer=2, activation function in the first hidden layer=symmetric saturating linear transfer function, activation function in the second hidden layer=symmetric saturating linear transfer function, and activation function in output layer=symmetric saturating linear transfer function, respectively.

Table. VI.2. The optimal structure for WFFNNs model based MISMO strategy using GA

Parametres	Model (t+1, t+2)	Model (t+3, t+4)
D	12	11
MWT	Db9	Db9
L	5	6
NHL1	9	2
NHL2	2	6
AFHL1	Satlins	Logsig
AFHL2	Satlins	Satlins
AFOL	Satlins	Satlins

Table (VI.3) provides the optimal structure of WFFNNs-GA model based on the MIMO strategy in this study. It can be judged from Table (VI.3) that the variables of WFFNNs-GA model with 1, 2, 3, and 4 day-ahead streamflow forecasting were suggested as follows: input delay=10, the type of mother wavelet=Sym7, decomposition level=4, number of neuron in first hidden layer=4, number of neuron in second hidden layer=6, activation function in the first hidden layer = log-sigmoid transfer function, activation function in the second hidden layer=symmetric saturating linear transfer function, and activation function in output layer=symmetric saturating linear transfer function, respectively. Tables (VI.1), (VI.2), and (VI.3) explain that the wavelet transform utilizing Daubechies 9 (Db9) contributed the selection of optimal model's structure based on three evolutionary strategies (i.e., MISO, MIMO, and MISMO) and improved the efficiency of the FFNNs model consequently. This finding followed the recent literature of Seo et al. (2015) closely. Also, GA determined all of parameters including wavelet transform that could yield the best performance without the intervention of human activity.

Table. VI.3. The optimal structure for WFFNNs model based MIMO strategy using GA

Parametres	Model (t+1, t+2, t+3, t+4)
D	10
MWT	Sym7
L	4
NHL1	4
NHL2	6
AFHL1	Logsig
AFHL2	Satlins
AFOL	Satlins

Table (VI.4) suggests the performance of WFFNNs-GA model based on the three evolutionary strategies (i.e., MISO, MIMO, and MISMO) for the multi-day-ahead (e.g., 1-, 2-, 3-, and 4-day-ahead) streamflow forecasting. For the multiday-ahead streamflow forecasting, the calibration and validation phases of WFFNNs-GA model, the forecasting results of the 1-day-ahead streamflow yielded the best performance based on RMSE, SNR, CC, NSE, and PFC indices for three evolutionary strategies (i.e., MISO, MIMO, and MISMO), respectively. As the multi-time-ahead streamflow forecasting for three evolutionary strategies (i.e., MISO, MIMO, and MISMO) were varied from 1- to 4-day-ahead one by one, it can be seen that the model accuracy decreased definitely. Also, it can be concluded that the 1-day-ahead streamflow forecasting can produce the least errors than the other multi-day-ahead (e.g., 2-, 3-, and 4-day-ahead) in this study. Table (VI.4) produces that the values of RMSE (e.g., 1.550, 4.659, 6.414, and 6.707 m³ /sec) and SNR (e.g., 0.066, 0.198, 0.273, and 0.285) indices for the WFFNNs-GA model based on the MISMO evolutionary strategy were lower than those of the WFFNNs-GA model based on the MISO and MIMO evolutionary strategies for the validation phase. In addition, the values of NSE and CC indices for the WFFNNs-GA model based on the MISMO

evolutionary strategy were higher than those of the WFFNNs-GA model based on the MISO and MIMO evolutionary strategies. It can be judged that the results of the WFFNNs-GA model based on the MISMO evolutionary strategy were superior to those of the WFFNNs-GA model based on the MISO and MIMO evolutionary strategies definitely. Since the structure of MISMO evolutionary strategy is more complex and complicated than that of MISO and MIMO evolutionary strategies, it is difficult to converge the global minimum errors utilizing few iterations numbers of MISMO evolutionary strategy for the calibration phase. However, the calibration phase utilizing the sufficient iterations number can improve the accuracy of WFFNNs-GA model utilizing MISMO strategy in this study.

Table. VI.4. Comparison between the performance results obtained by the MISO, MIMO, and MISMO strategies in the training and testing phases for WFFNNs models.

		Training phase					Testing phase				
		RMSE	NSR	NSE	R	PFC	RMSE	NSR	NSE	R	PFC
		(m ³ /s)		(%)			(m ³ /s)		(%)		(m ³ /s)
MISO	t+1	1.491	0.047	99.783	0.999	0.053	1.576	0.067	99.550	0.998	0.072
	t+2	3.687	0.115	98.672	0.994	0.186	6.820	0.290	91.579	0.959	0.199
	t+3	6.584	0.206	95.766	0.979	0.201	7.553	0.321	89.670	0.947	0.190
	t+4	5.222	0.163	97.337	0.987	0.052	24.355	1.036	-7.398	0.714	0.315
MIMO	t+1	3.389	0.106	98.878	0.995	0.206	3.448	0.147	97.848	0.989	0.121
	t+2	6.232	0.195	96.206	0.981	0.229	6.236	0.265	92.958	0.966	0.179
	t+3	7.441	0.233	94.592	0.973	0.245	7.660	0.326	89.376	0.953	0.198
	t+4	6.520	0.204	95.847	0.979	0.169	11.017	0.469	78.023	0.883	0.250
MISMO	t+1	2.047	0.064	99.591	0.998	0.131	1.550	0.066	99.565	0.998	0.081
	t+2	6.805	0.213	95.477	0.978	0.179	4.659	0.198	96.070	0.981	0.135
	t+3	5.573	0.174	96.967	0.985	0.199	6.414	0.273	92.552	0.962	0.172
	t+4	5.913	0.185	96.585	0.983	0.136	6.707	0.285	91.856	0.958	0.164

Table (VI.5) shows the performance of FFNNs-GA models based on the three evolutionary strategies (i.e., MISO, MIMO, and MISMO) for the multi-day-ahead (e.g., 1-, 2-, 3-, and 4-day-ahead) streamflow forecasting. For the calibration and validation phases of FFNNs-GA model based on the multi-day-ahead forecasting, the results of FFNNsGA model based on the MISO evolutionary strategy were superior to those of FFNNs-GA model based on the MIMO and MISMO evolutionary strategies for 1- and 2-day-ahead forecasting. However, the results of FFNNs-GA model based on the MISMO evolutionary strategy were superior to those of FFNNs-GA model based on the MISO and MIMO evolutionary strategies for 3 and 4 day

ahead forecasting. In addition, since the values of NSE index in the FFNNs-GA model based on the MIMO evolutionary strategy produced the negative values, they presented the unacceptable performance in the validation phase.

Table. VI.5. Comparison between the performance results obtained by the MISO, MIMO and MISMO strategies in the training and testing phases for FFNNs models.

	Training phase					Testing phase					
	RMSE	NSR	NSE	R	PFC	RMSE	NSR	NSE	R	PFC	
	(m ³ /s)		(%)			(m ³ /s)		(%)		(m ³ /s)	
MISO	t+1	17.113	0.535	71.395	0.846	0.446	11.486	0.489	76.113	0.879	0.243
	t+2	21.302	0.666	55.676	0.747	0.506	15.301	0.651	57.610	0.760	0.290
	t+3	27.481	0.859	26.235	0.513	0.657	17.727	0.754	43.100	0.685	0.298
	t+4	22.592	0.706	50.149	0.715	0.451	39.643	1.686	-184.5	0.322	0.314
MIMO	t+1	17.819	0.557	68.986	0.834	0.404	59.780	2.543	-547.0	0.593	0.540
	t+2	19.279	0.602	63.695	0.800	0.348	68.370	2.908	-746.3	0.458	0.525
	t+3	24.518	0.766	41.284	0.647	0.555	37.009	1.574	-147.9	0.488	0.377
	t+4	25.872	0.808	34.623	0.596	0.505	45.122	1.919	-268.6	0.435	0.389
MISMO	t+1	17.389	0.543	70.463	0.841	0.395	22.847	0.972	5.485	0.814	0.350
	t+2	24.370	0.762	41.990	0.648	0.582	20.763	0.883	21.941	0.717	0.311
	t+3	27.508	0.860	26.090	0.511	0.651	17.106	0.728	47.018	0.693	0.300
	t+4	27.809	0.869	24.465	0.496	0.642	17.736	0.754	43.043	0.657	0.305

Comparison between Tables (VI.4) and (VI.5) revealed that the results of WFFNNs-GA model were superior to those of FFNNs-GA for corresponding evolutionary strategy, respectively. In addition, because sub-time series captured the high variation that existed into the original series of inputs data, it can be seen that utilizing sub-time series decomposed by

DWT method as input data of WFFNNs-GA model can improve the performance of FFNNs-GA model clearly.

Figure (VI.4) presents the scatter diagrams for the WFFNNsGA model based on the evolutionary MISO, MIMO, and MISMO strategies, respectively. The straight line in Fig(VI.4) represents best-ft line equation. It was clearly seen from the best-ft line and R^2 values that the WFFNNs-GA model with 1-day-ahead based on the three evolutionary strategies (i.e., MISO, MIMO, and MISMO) could forecast the daily streamfow better than other multi-step-ahead (e.g., 2, 3, and 4 day ahead) forecasting clearly. It can be also judged that the WFFNNs-GA model based on the MISMO evolutionary strategy can forecast the daily streamfow better than the corresponding WFFNNs-GA model based on the MISO and MIMO evolutionary strategies from the viewpoint of the best-ft line and R^2 values.

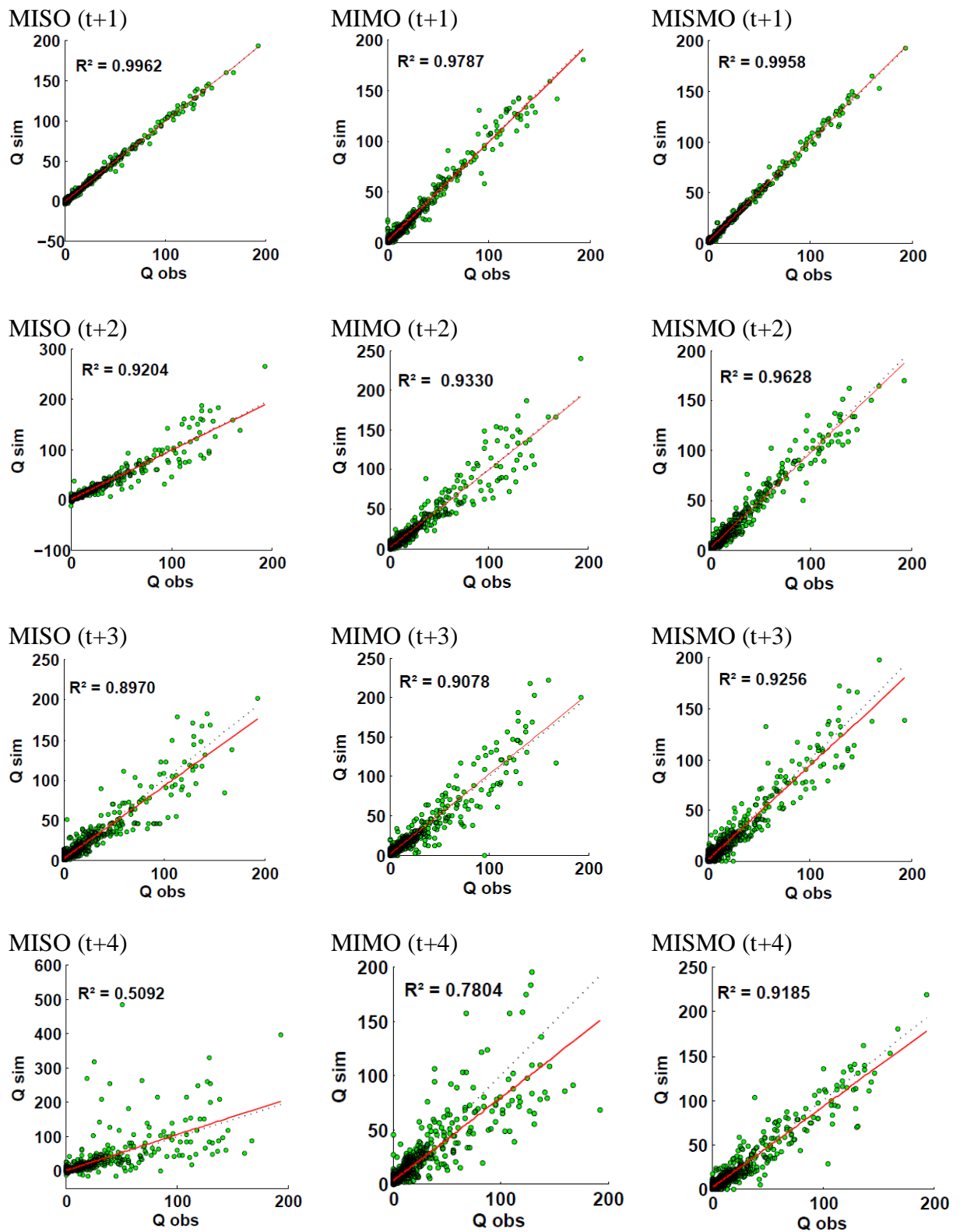


Figure. VI.4. Scatter diagram for WFFNNs models using MISO, MIMO, and MIMO strategies (Testing phase)

Figure (VI.5) produces the scatter diagrams for the FFNNsGA model based on the evolutionary MISO, MIMO, and MISMO strategies, respectively. It can be seen from the bestfit line and R^2 values that the FFNNs-GA model with 1 and 2 day ahead based on the MISO evolutionary strategy can forecast the daily streamflow better than the corresponding FFNNs models based on the MIMO and MISMO evolutionary strategies. It can be also derived, however, that FFNNs-GA model with 3 and 4 day-ahead utilizing the MISMO evolutionary strategy can forecast the daily streamflow better than the corresponding FFNNs-GA models based on the MISO and MIMO evolutionary strategies from the viewpoint of the best-fit line and R^2 values, respectively. In the current research, the authors developed a heuristic model combined a data pre-processing technique and an optimization algorithm based on three evolutionary strategies (i.e., MISO, MIMO, and MISMO).

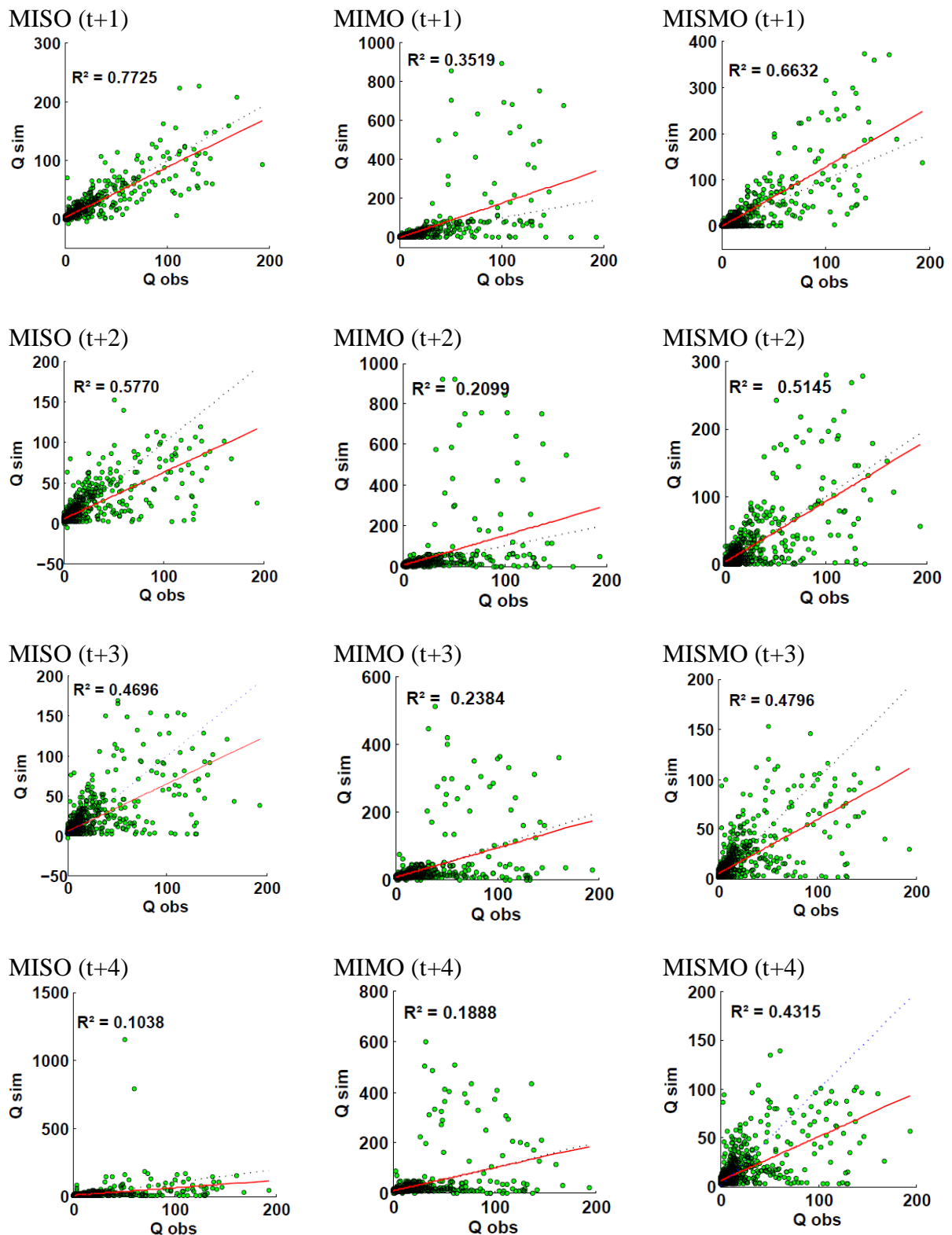


Figure. VI.5. Scatter diagram for FFNNs models using MISO, MIMO, and MISMO strategies (Testing phase)

It can be found that this combination can be applied the complex and non-stationary natural phenomenon including the nonlinear hydrologic time series (e.g., rainfall, groundwater, evaporation, and water stage etc.). In addition, an addressed attempt to forecast the daily streamflow based on three evolutionary strategies (i.e., MIMO, MISO, and MISMO) is unique process compared to the previous researches following the category of multi-day-ahead streamflow forecasting (Seo et al. 2018; Zakhrouf et al. 2018; Abdollahi et al. 2017; Ravansalar et al. 2016, 2017; Zakhrouf et al. 2016; Kalteh 2015; Seo et al. 2015; Nourani et al. 2014; Sahay and Srivastava 2014). The forecasted results can be accepted based on the diverse statistical indices and scatter diagrams. In addition, the various heuristic models (e.g., multilayer perceptron (MLP), generalized regression neural networks (GRNNs), support vector machines (SVMs), extreme learning machines (ELMs), multivariate adaptive regression spline (MARS), and genetic programming (GP) etc.) combined data pre-processing techniques and optimization algorithms based on evolutionary strategies can be proposed to confirm the accuracy and efficiency of multistep-ahead streamflow forecasting for the further researches.

VI.8. Conclusion

This study suggests the wavelet-based feed forward neural networks (WFFNNs) model optimized utilizing genetic algorithm (GA) based on the three evolutionary strategies [i.e., multi-input multi-output (MIMO), multi-input single-output (MISO), and multi-input several multi-output (MISMO)], for streamflow forecasting in the Chellif River basin, Algeria. The WFFNNs-GA model based on the three evolutionary strategies evaluates the forecasting accuracy for multi-step-ahead (e.g., one, two, three, and four step ahead) streamflow utilizing five statistical indices including root mean squared error (RMSE), signal-to-noise ratio (SNR), correlation coefficient (R), coefficient of efficiency (NSE), and peak flow criteria (PFC). The

optimal structure of WFFNNs-GA model is categorized utilizing the input delay, the type of mother wavelet, decomposition level, number of neuron in hidden layers, and activation functions in the hidden, and output layers, respectively. Also, the optimal values of each variable are determined utilizing the calibration dataset.

It can be concluded that the WFFNNs-GA model based on the three evolutionary strategies produces the best results with 1 day ahead streamflow forecasting. Within the category of 1-day-ahead streamflow forecasting, the WFFNNsGA model based on the MISO evolutionary strategy provides the values of RMSE=1.576 (m³ /sec), SNR=0.067, R=0.998, NSE=99.550 (%), and PFC=0.072 for the validation phase. In the MIMO evolutionary strategy, it gives the values of RMSE=3.443 (m³ /sec), SNR=0.146, R=0.990, NSE=97.853 (%), and PFC=0.119 for the validation phase. In addition, it also yields the values of RMSE=1.550 (m³ / sec), SNR = 0.066, R = 0.998, NSE = 99.565 (%), and PFC=0.081 within the MISMO evolutionary strategy for the validation phase. It can be suggested from the WFFNNsGA model that the results of MISMO evolutionary strategy are superior to those of MISO and MIMO evolutionary strategies based on the statistical indices and scatter diagrams. The WFFNNs-GA model based on the MISMO evolutionary strategy can forecast the accurate and efficient streamflow compared with the other WFFNNs-GA models based on the MISO and MIMO evolutionary strategies in this study. Results also indicate that the application of sub-time series decomposed by DWT as the input data of WFFNNsGA model can improve the performance of FFNNs-GA models, and forecast the accurate streamflow in Chellif River basin, Algeria.

General conclusion

Within this thesis, an effort is made to build highly reliable flow forecasting models using advanced models of artificial neural networks. Unlike conventional model-based physical methods, time series models do not require a number of variables to model dynamic hydrological processes.

Time series models are capable of extracting information from only previous values in order to achieve accurate future values. The implementation of numerous models of artificial neural networks, including feed forward neural networks, adaptive neuro-fuzzy inference systems, hybrid wavelet neural networks and recurrent neural networks models, is investigated.

Five different catchements with different characteristics are selected as case studies; Ain Safra bassin in south Algeria, Seybouss bassin, Soummam bassin, Sebaou bassin and Chellif bassin in north of Algeria.

Chapter One of this thesis provides background material and theoretical descriptions of artificial neural networks. Each of the remaining chapters (2 to 6) of this thesis deals with a case study of one or more watersheds.

Chapter Two present an application of the wavelet based feed forward neural networks model (WFFNN), adaptive Neuro-Fuzzy Inference systems (ANFIS) and the simple feed forward neural network (FFNN) for modeling the streamflow of a catchment located in the south west of Algeria namely Ain Safra watershed. This catchment is prone to a semi-arid climate and a strong variability in runoff. The time series of its daily rainfall–runoff are used for our models. In the first model (WFFNN), the time series of rainfall and streamflow are decomposed into a

succession of approximation and details using the discrete wavelet transform and used as inputs for the feed forward neural network. The second model corresponds to Adaptive Network-based Fuzzy Inference System (ANFIS), which generates an input–output based on both fuzzy rules and stipulated rainfall–runoff data pairs. The obtained results show that the performances of (WFFNN) and (ANFIS) models exceed those of (FFNN) model.

In chapter Three, an effort has been made to develop a conjunction model – wavelet transformation, artificial neural networks models, and genetic algorithm (GA) – for forecasting the daily streamflow of a river in northern center of Algeria using the time series of runoff. This catchment represented by the Sebaou watershed. The original time series was decomposed into multi-frequency time series by wavelet transform algorithm and used as inputs to feed forward neural network (FFNN) and adaptive neuro-fuzzy inference system (ANFIS) models. Several factors must be optimized to determine the best model structures. Wavelet based artificial neural networks models using a (GA) are designed to optimize model structure. The performances of wavelet-based artificial neural networks models (i.e. WANFIS and WFFNN) were superior to those of conventional models.

Chapter Four investigated four artificial neural networks models (i.e., Elman recurrent neural network (ERNN), long shortterm memory (LSTM), and gated recurrent unit (GRU)) models to forecast streamflow at Sidi Aich and Ponteba Defluent stations, represented by Soummam and Chellif waterheds respectively, Algeria. Also, feedforward neural network (FFNN) was implemented to compare the accuracy of our models. The particle swarm optimization (PSO), one of popular evolutionary optimization methods, was combined to determine the automated optimal hyperparameters. The results shows that the (GRU) model with two-day-lag forecasted the daily streamflow accurately compared to other models at Sidi Aich station.

Also, (GRU) model with two-day-lag performed the daily streamflow forecasting efficiently at Ponteba Defluent station.

Chapter Five aims to develop and apply three different artificial neural networks approaches (i.e., feed forward neural networks (FFNN), adaptive neuro-fuzzy inference systems (ANFIS), and wavelet-based feed forward neural networks (WFFNN)) combined with an evolutionary optimization algorithm and the k-fold cross validation for multi-step (days) streamflow forecasting at the catchment of Seybouss watershed located in north est of Algeria. The results performance criteria for the WFFNN model were best than those of FFNN and ANFIS models.

In the last chapter (six), the feed forward neural networks (FFNNs) were proposed to forecast the multi-day-ahead streamflow for the Chellif River, Algeria. The parameters of FFNNs model were optimized utilizing genetic algorithm (GA). Moreover, discrete wavelet transform was utilized to enhance the accuracy of FFNNs model's forecasting. Therefore, the wavelet-based feedforward neural networks (WFFNNs-GA) model was developed for the multi-day-ahead streamflow forecasting based on three evolutionary strategies [i.e., multi-input multi-output (MIMO), multi-input single-output (MISO), and multi-input several multi-output (MISMO)]. Results provided that the statistical values of WFFNNs-GA model based on MISMO evolutionary strategy were superior to those of WFFNNs-GA model based on (MISO) and (MIMO) evolutionary strategies for the multi-day-ahead streamflow forecasting. Results indicated that the performance of (WFFNNs-GA) model based on (MISMO) evolutionary strategy provided the best accuracy. Results also explained that the hybrid model suggested better performance compared with stand-alone model based on the corresponding

evolutionary strategies. Therefore, the hybrid model can be an efficient and robust implement to forecast the multi-day-ahead.

Finally we can summarized conclusions of this study as Following;

- In modelling and forecasting of short term (one-step ahead) or long term (multi-step ahead) streamflow, time series models based artificial neural networks are found to be very promising alternative.
- The wavelet based artificial neural networks hybrid models are often outperformed then traditional models of artificial neural networks for both short term and long term forecasting.
- In the case of recurrent neural networks this study proved that the GRU model gives the highest accuracy to forecast streamflow time series in Algeria.
- The MISMO strategie was found to outperform to forecast multi step ahead streamflow.

References

- Abdollahi, S., Raeisi, J., Khalilianpour, M., Ahmadi, F., and Kisi, O. (2017). "Daily mean streamflow prediction in perennial and non-perennial rivers using four data driven techniques." *Water Resources Management*, Vol. 31, No. 15, pp. 4855-4874.
- Abrahart, R.J., and See, L. (2000). "Comparing neural network and autoregressive moving average techniques for the provision of continuous river flow forecasts in two contrasting catchments." *Hydrological Processes*, Vol. 14, No. 11-12, pp. 2157-2172.
- Aichouri, I., Hani A., Bougherira, N., Djabri, L., Chaffai, H., and Lallahem, S. (2015). "River flow model using artificial neural networks." *Energy Procedia*, Vol. 74, pp. 1007-1014.
- Allili-Ailane, C., Laignel, B., Adjeroud, N., Bir, H., and Madania, K. (2015). Particulate Flow at the Mouth of the Soummam Watershed (Algeria). *Environmental Progress & Sustainable Energy*, DOI 10.1002/ep.
- Alpaslan, Y., Onucyıldız, M., Coptu, N.K. (2009). Modelling Level Change In Lakes Using Neuro-Fuzzy And Artificial Neural Networks, *J. Hydrol*, 365, 329–334.
- 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 behavior of runoff, *J. Hydrol*, 337, 22-34.
- Arlot, S., Celisse, A. (2010). Cross-validation procedures for model selection. *Statistics Surveys*, 4, 40–79.
- Asadi, S., Shahrabi, J., Abbaszadeh, P., Tabanmehr, S. (2013). A new hybrid artificial neural networks for rainfall–runoff process modeling. *Neurocomputing*, 121, 470-480.
- ASCE Task Committee on Definition of Criteria for Evaluation of Watershed Models of the Watershed Management Committee, Irrigation and Drainage Division. (1993). Criteria for evaluation of watershed models. *Journal of Irrigation and Drainage Engineering*, 119(3), 429-442.
- Badrzadeh, H., Sarukkalige, R., Jayawardena, A.W. (2013). Impact of multi-resolution analysis of artificial intelligence models inputs on multi-step ahead river flow forecasting. *Journal of Hydrology*, 507, 75-85.
- Baldominos, A., Saez, Y., Isasi, P. (2018). Evolutionary convolutional neural networks: An application to handwriting recognition. *Neurocomputing*, 283, 38-52.
- Benhattab, K., Bouvier, C., Meddi, M. (2011). Regional analysis for the estimation of low-frequency daily rainfalls in chelif catchment, Algeria, *Proceedings of the MED-FRIEND International Workshop on Hydrological Extremes*, held at University of Calabria, Cosenza (Italy).
- Ben Taieb, S., Sorjamaa, A., Bontempi, G. (2010). Multiple-output modeling for multi-step-ahead time series forecasting. *Neurocomputing*, 73, 1950 – 1957.

- Biswas, R.K., and Jayawardena, A.W. (2014). "Water level prediction by artificial neural network in a flashy transboundary river of Bangladesh." *Global Nest Journal*, Vol. 16, No. 2, pp. 432-444.
- Bouchelkia, H., Belarbi, F., Remini, B. (2014). Quantification of suspended sediment load by double correlation in the watershed of Chellif (Algeria). *Journal of Water and Land Development*, 21, 39–46.
- Bouktif, S., Fiaz, A., Ouni, A., Serhani, M.A. (2020). Multi-sequence LSTM-RNN deep learning and metaheuristics for electric load forecasting. *Energies*, 13(2), 391.
- Bormann, H. (2005). Evaluation of hydrological models for scenario analyses: signal-to-noise-ratio between scenario effects and model uncertainty. *Advances in Geosciences*, European Geosciences Union, 43-48.
- Cigizoglu, H.K. (2003). "Estimation, forecasting and extrapolation of river flows by artificial neural networks." *Hydrological Sciences Journal*, Vol. 48, No. 3, pp. 349-361.
- Chandwani, V., Vyas, S.K., Agrawal, V., and Sharma, G. (2015). "Soft computing approach for rainfall-runoff modelling: A review." *Aquatic Procedia*, Vol. 4, No. 2015, pp. 1054-1061.
- Chang, F.J., Chiang, Y.M., Chang, L.C. (2007). Multi-step-ahead neural networks for flood forecasting. *Hydrological Sciences Journal*, 52, 114-130.
- Chen, D., Wang, J., Zou, F., Yuan, W., Hou, W. (2014). Time series prediction with improved neuro-endocrine model. *Neural Computing and Applications*, 24(6), 1465-1475.
- Chen, L., Li, Z., Zhang, Y. (2019). Multiperiod-ahead wind speed forecasting using deep neural architecture and ensemble learning. *Mathematical Problems in Engineering*, 2019.
- Cheng, G., Zhang, P., Xu, J. (2019). Automatic speech recognition system with output-gate projected gated recurrent unit. *IEICE Transactions on Information and Systems*, 102(2), 355-363.
- Cheng, Y., Xu, C., Mashima, D., Thing, V.L., Wu, Y. (2017). PowerLSTM: power demand forecasting using long short-term memory neural network. In *International Conference on Advanced Data Mining and Applications*, Springer, Cham, 727-740.
- Chettih, M., Zakhrouf, M. (2012). Performance of artificial intelligence algorithm hybridization to wavelet transforms and fuzzy logic in the forecast of extreme flow, *JHYMER*, 3, 14-22.
- Cho, K., Van Merriënboer, B., Bahdanau, D., Bengio, Y. (2014). On the properties of neural machine translation: Encoder-decoder approaches. *arXiv preprint arXiv:1409.1259*.
- Chouaib, M. (2016). Impact of Wavelet Transformation on Data Driven Rainfall-Runoff Models, doctorat thesis, University of Auckland.
- Chu, H., Wei, J., Li, T., Jia, K. (2016). Application of support vector regression for mid- and long-term runoff forecasting in "Yellow river headwater" region. *Procedia Engineering* 154, 1251 – 1257.

Dadu, K. S., Deka, P. C. (2013). Multistep lead time forecasting of hydrologic time series using daubechies wavelet. *International Journal of Scientific & Engineering Research*, 4 (10), 115- 124.

Danandeh Mehr, A., Kahya, E., Olyaie, E. (2013). Streamflow prediction using linear genetic programming in comparison with a neuro-wavelet technique. *Journal of Hydrology*, 505, 240-249.

Dariane, A.B., Azimi, S. (2016). Forecasting streamflow by combination of a genetic input selection algorithm and wavelet transforms using ANFIS models. *Hydrological Sciences Journal*, 61(3), 585-600.

Dastorani, M., Moghadamnia, A., Piri, J., Rico-Ramirez, M. (2010). Application of ANN and ANFIS models for reconstructing missing flow data, *Environ. Monit. Assess*, 166, 421-434.

Delafrouz, H., Ghaheri, A., Ghorbani, M.A. (2017). A novel hybrid neural network based on phase space reconstruction technique for daily river flow prediction. *Soft Computing*, 1-11.

Derdour A., Bouanani A., Babahamed K. (2016). Floods typology in semiarid environment: case of ain sefra watershed (ksour mountains, saharian atlas, sw of algeria), *Larhyss Journal*, 29, 283-299.

Ding, S., Li, H., Su, C., Yu, J., Jin, F. (2013). Evolutionary artificial neural networks: a review. *Artificial Intelligence Review*, 39, 251-260.

Dong, X., Wang, S., Sun, R., Zhao, S. (2010). Design of artificial neural networks using a genetic algorithm to predict saturates of vacuum gas oil. *Petroleum Science*, 7, 118-122.

Droogers, P., & Aerts, J., (2005). Adaptation strategies to climate change and climate variability: a comparative study between seven contrasting river basins. *Physics and Chemistry of the Earth*, 30: 339 – 346. In : Droogers P et al. (2011).

Droogers, P., Terink, W., Hunink, J., Kauffman, S., & Van Lynden, G., (2011). Green Water Management Options in the Sebou Basin: Analysing the Costs and Benefits using WEAP. GWC Report M2b.

Droogers, P., & Bastiaanssen, W., (2002). Irrigation Performance using Hydrological and Remote Sensing Modeling. *J. Irrig. Drain Eng.*, 128(1), 11–18. In : Droogers P et al. (2011).

Eberhart, R., Kennedy, J. (1995). A new optimizer using particle swarm theory. In *MHS'95. Proceedings of the Sixth International Symposium on Micro Machine and Human Science*, IEEE, 39-43.

Elman, J.L. (1990). Finding structure in time. *Cognitive science*, 14(2), 179-211.

Elmasry, W., Akbulut, A., Zaim, A.H. (2020). Evolving deep learning architectures for network intrusion detection using a double PSO metaheuristic. *Computer Networks*, 168, 107042.

ErolKeskin, M., Taylan, D., Terzi, Ö. (2010). Adaptive neural-based fuzzy inference system (ANFIS) approach for modeling hydrological time series, *Hydrol.Scie.J.*, 51.4, 588-598.

- Fielding, B., Zhang, L. (2018). Evolving image classification architectures with enhanced particle swarm optimisation. *IEEE Access*, 6, 68560-68575.
- Fortin ,V., Ouarda, T.B.M.J., Rasmussen, P.F. and Bobée, B. (1997). A review of stream flow forecasting methods, *Rev. Sci. Eau*, 4, 461-487.
- Fredric, J. (2007). Modèle de prévision et de gestion des crues - Optimisation des opérations des aménagements hydroélectriques à accumulation pour la réduction des débits de crue, ISSN 1661-1179.
- Fu, M., Fan, T., Ding, Z.A., Salih, S.Q., Al-Ansari, N., Yaseen, Z.M. (2020). Deep learning data-intelligence model based on adjusted forecasting window scale: Application in daily streamflow simulation. *IEEE Access*, 8, 32632-32651.
- Gasparin, A., Lukovic, S., Alippi, C. (2019). Deep learning for time series forecasting: The electric load case. *arXiv preprint arXiv:1907.09207*.
- Gautam, D.K., Holz, K.P. (2001). Rainfall-runoff modeling using adaptive neuro-fuzzy systems,*J.Hydroinf*, 03.1,3-10.
- Gökgöz, F., Filiz, F. (2018). Deep learning for renewable power forecasting: An approach using LSTM neural networks. *International Journal of Energy and Power Engineering*, 12(6), 416-420.
- Gowda ,C.C., Mayya, S.G. (2014). Comparison of Back Propagation Neural Network and Genetic Algorithm Neural Network for Stream Flow Prediction. *Journal of Computational Environmental Sciences*, doi.org/10.1155/2014/290127.
- Guo, J., Zhou, J., Qin, H., Zou, Q., Li, Q. (2011). Monthly streamflow forecasting based on improved support vector machine model. *Expert Systems with Applications*, 38(10), 13073-13081.
- Haykin, S. (1999). *Neural networks: a comprehensive foundation*, Prentice-Hall, Upper Saddle River, N.J., 178-274.
- Hochreiter, S., Schmidhuber, J. (1997). Long short-term memory. *Neural computation*, 9(8), 1735-1780.
- Holland, J.H. (1975). *Adaptation in natural and artificial systems*. University of Michigan Press. (Second edition: MIT Press, 1992.)
- Hu, Y., Sun, X., Nie, X., Li, Y., Liu, L. (2019). An enhanced LSTM for trend following of time series. *IEEE Access*, 7, 34020-34030.
- Imrie, C.E., Durucan, S., and Korre, A. (2000). "River flow prediction using artificial neural networks: Generalisation beyond the calibration range." *Journal of Hydrology*, Vol. 233, No. 1, pp. 138-153.

- Jang, J. S. R. (1993). ANFIS: Adaptive-network-based fuzzy inference system. *IEEE Transactions on Systems, Man, and Cybernetics*, 23(3), 665–685.
- Kalteh, A.M. (2015). Wavelet genetic algorithm-support vector regression (wavelet GA-SVR) for monthly flow forecasting. *Water Resources Management*, 29, 1283-1293.
- Kamruzzaman, M., Metcalfe, A.V., and Beecham, S. (2013). “Waveletbased rainfall-stream flow models for the southeast Murray Darling basin.” *Journal of Hydrologic Engineering*, Vol. 19, No. 7, pp. 1283-1293.
- Keskin, M.E., Taylan, D., and Terzi, Ö. (2006). “Adaptive neural-based fuzzy inference system (ANFIS) approach for modelling hydrological time series.” *Hydrological Sciences Journal*, Vol. 51, No. 4, pp. 588- 598.
- Khodayar, M., Kaynak, O., Khodayar, M.E. (2017). Rough deep neural architecture for short-term wind speed forecasting. *IEEE Transactions on Industrial Informatics*, 13(6), 2770-2779.
- Kim, S., Kim, H.S. (2008). Uncertainty reduction of the flood stage forecasting using neural networks model. *JAWRA Journal of the American Water Resources Association*, 44(1), 148-165.
- Kimura, N., Yoshinaga, I., Sekijima, K., Azechi, I., Baba, D. (2020). Convolutional neural network coupled with a transfer-learning approach for time-series flood predictions. *Water*, 12(1), 96.
- Kingma, D.P., Ba, J. (2014). Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980.
- Kisi, O., Shiri, J. (2012). Reply to discussion of “Precipitation forecasting using wavelet-genetic programming and wavelet-neuro-fuzzy conjunction models”. *Water Resources Management*, 26(12), 3663-3665.
- Kline, D.M. (2004). Methods for multi-step time series forecasting with neural networks. *Neural Networks in Business Forecasting*, 226-250.
- Komori, Y. (1992). A neural fuzzy training approach for continuous speech recognition improvement. In *ICASSP-92: 1992 IEEE international conference on acoustics, speech and signal processing* (Vol. 1, pp. 405–408). IEEE.
- Koudstaal, R., Rijsberman, F. R., Savenije, H. (1992). Water and sustainable development. *Natural Resources Forum*, Doi: 0165-0203/92, 040277-1.
- Kratzert, F., Klotz, D., Brenner, C., Schulz, K., Herrnegger, M. (2018). Rainfall–runoff modelling using long short-term memory (LSTM) networks. *Hydrology and Earth System Sciences*, 22(11), 6005-6022.
- Krishna, B., Satyaji Rao, Y.R., Nayak, P.C. (2011). Time series modeling of river flow using wavelet neural networks. *Journal of Water Resource and Protection*, 3, 50-59.

- Labat, D., Ababou, R., & Mangin, A. (2000). Rainfall–runoff relations for karstic springs. Part II: Continuous wavelet and discrete orthogonal multiresolution analyses. *Journal of Hydrology*, 238, 149–178.
- Lawrance, A.J., Kottegoda, N.T. (1977). Stochastic modelling of riverflow time series. *Journal of the Royal Statistical Society: Series A (General)*, 140(1), 1-31.
- Le, X.H., Ho, H.V., Lee, G., Jung, S. (2019). Application of long short-term memory (LSTM) neural network for flood forecasting. *Water*, 11(7), 1387.
- Legates, D.R., McCabe Jr, G.J. (1999). Evaluating the use of “goodness-of-fit” measures in hydrologic and hydroclimatic model validation. *Water Resources Research*, 35(1), 233-241.
- Liu, D., Niu, D., Wang, H., Fan, L. (2014). Short-term wind speed forecasting using wavelet transform and support vector machines optimized by genetic algorithm. *Renewable Energy*, 62, 592-597.
- Liu, M., Huang, Y., Li, Z., Tong, B., Liu, Z., Sun, M., Jiang, F., Zhang, H. (2020). The applicability of LSTM-KNN model for real-time flood forecasting in different climate zones in China. *Water*, 12(2), 440.
- Liu, W.C., and Chung, C.E. (2014). “Enhancing the predicting accuracy of the water stage using a physical-based model and an artificial neural network-genetic algorithm in a river system.” *Water*, Vol. 6, No. 6, pp. 1642-1661.
- Lohani, A. K., Goel, N. K., & Bhatia, K. K. S. (2006). Takagi-Sugeno fuzzy inference system for modeling stage–discharge relationship. *Journal of Hydrology*, 331, 146–160.
- Mallat, S. G. (1989). A theory for multiresolution signal decomposition: The wavelet representation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 11, 674–693.
- Mangin, A. (1984). The use of autocorrelation and spectral analyses to obtain a better understanding of hydrological systems, *J.Hydrol*, 67, 25-43.
- Martins, O.Y., Sadeeq, M.A., Ahaneku, I.E. (2011). ARMA modelling of Benue river flow dynamics: comparative study of PAR model. *Open Journal of Modern Hydrology*, 1, 1-9.
- McCulloch, W. S., & Pitts, W. (1943). A logical calculus of the ideas immanent in nervous activity. *The Bulletin of Mathematical Biophysics*, 5, 115–133.
- Moosavi, V., Vafakhah, M., Shirmohammadi, B., & Behnia, N. (2013). A wavelet-ANFIS hybrid model for groundwater level forecasting for different prediction periods. *Water Resources Management*, 27, 1301–1321.
- Moradkhani, H., Hsu, K.L., Gupta, H.V., and Sorooshian, S. (2004). “Improved streamflow forecasting using self-organizing radial basis function artificial neural networks.” *Journal of Hydrology*, Vol. 295, No. 1, pp. 246-262.

Moriasi, D.N., Arnold, J.G., Van Liew, M.W., Bingner, R.L., Harmel, R.D., Veith, T.L. (2007). Model evaluation guidelines for systematic quantification of accuracy in watershed simulations. *Transactions of the ASABE*, 50(3), 885-900.

Nash, J.E. and Sutcliffe, J.V. (1970). River flow forecasting through conceptual models I: a discussion of principles. *Journal of Hydrology*, 10, 282–290.

Nason GP (2008) *Wavelet methods in statistics with R*. Springer Science and Business Media, New York, pp 15–78. <https://doi.org/10.1007/978-0-387-75961-6>.

Nasr, A., Bruen, M. (2008). Development of neuro-fuzzy models to account for temporal and spatial variations in a lumped rainfall-runoff model, *J. Hydrol*, 349, 277-290.

Nayak, P.C., Shudheer, K.P., Ranganand, D.M., Ramasastri, K.S. (2004). A neuro-fuzzy computing technique for modeling hydrological time series, *J. hydrol*, 2952-66.

Nourani V, Alami MT, Aminfar MH (2009) A combined neural-wavelet model for prediction of Ligvanchai watershed precipitation. *Eng Appl Artif Intell* 22(3):466–472.

Nourani, V., Hosseini, B.A., Adamowski, J., Gebremicheal, M. (2013). Using self-organizing maps and wavelet transforms for space-time pre-processing of satellite precipitation and runoff data in neural network based rainfall–runoff modeling. *Journal of Hydrology*, 476, 228–243.

Nourani, V., Baghanam, A.H., Adamowski, J., Kisi, O. (2014). Applications of hybrid wavelet–Artificial Intelligence models in hydrology: A review. *Journal of Hydrology*, 514, 358-377.

Parmar, K.S., Bhardwaj, R. (2015). River water prediction modeling using neural networks, fuzzy and wavelet coupled model. *Water Resources Management*, 29, 17–33.

Partal, T. (2009). Modelling evapotranspiration using discrete wavelet transform and neural networks. *Hydrological processes*, 23(25), 3545-3555.

Partal, T., Kisi, O. (2007). Wavelet and neuro-fuzzy conjunction model for precipitation forecasting. *Journal of Hydrology*, 342, 199–212.

Prahlada, R., & Deka, P. C. (2011). Hybrid wavelet neural network model for improving forecasting accuracy of time series significant wave height. *International Journal of Earth Sciences and Engineering*, 4(5), 857–866.

Prechelt, L. (1998). Automatic Early Stopping Using Cross Validation: Quantifying the Criteria. *Neural Networks*, 11, 761-767.

Qiu, J., Wang, B., Zhou, C. (2020). Forecasting stock prices with long-short term memory neural network based on attention mechanism. *PloS ONE*, 15(1), e0227222.

Ravansalar, M., Rajaei, T., and Kisi, O. (2017). “Wavelet-linear genetic programming: A new approach for modeling monthly streamflow.” *Journal of Hydrology*, Vol. 549, pp. 461-475.

Ravansalar, M., Rajae, T., Zounemat-Kermani, M. (2016). A wavelet–linear genetic programming model for sodium (Na⁺) concentration forecasting in rivers. *Journal of Hydrology*, 537, 398-407.

Ren, G., Cao, Y., Wen, S., Huang, T., Zeng, Z. (2018). A modified Elman neural network with a new learning rate scheme. *Neurocomputing*, 286, 11-18.

Rezaie-Balf, M., Kim, S., Fallah, H., and Alaghmand, S. (2019). “Daily river flow forecasting using ensemble empirical mode decomposition based heuristic regression models: Application on the perennial rivers in Iran and South Korea.” *Journal of Hydrology*, Vol. 572, pp. 470-485.

Riad, S. (2004). *Typologie et analyse hydrologique des eaux superficielles a partir de quelques bassins versants representatifs du maroc*, doctorat thesis, universite ibnou zohr d’agadir.

Rio Conference (1992). <https://www.un.org/en/conferences/environment/rio1992>.

Sahay, R.R., Srivastava, A. (2014). Predicting monsoon floods in rivers embedding wavelet transform, genetic algorithm and neural network. *Water Resources Management*, 28, 301-317.

Salas, J.D. (1993). Analysis and modelling of hydrological time series. *Handbook of hydrology*, 19.

Samsudin, R., Saad, P., Shabri, A. (2011). River flow time series using least squares support vector machines. *Hydrology and Earth System Sciences*, 15(6), 1835-1852.

Seo, Y., Kim, S.(2016). River stage forecasting using wavelet packet decomposition and data-driven Models. *Procedia Engineering*, 154, 1225-1230.

Seo, Y., Kim, S., and Singh, V. (2018). “Machine learning models coupled with variational mode decomposition: A new approach for modeling daily rainfall-runoff.” *Atmosphere*, Vol. 9, No. 7, p. 251.

Seo, Y., Kim, S., Kisi, O., Singh, V.P. (2015a). Daily water level forecasting using wavelet decomposition and artificial intelligence techniques, *Journal of Hydrology*, 520, 224-243.

Seo, Y., Kim, S., Singh, V.P. (2015b). Multistep-ahead flood forecasting using wavelet and data-driven methods. *KSCE Journal of Civil Engineering*, 19(2), 401-417.

Seo, Y., Kim, S., Kisi, O., Singh, V.P., Parasuraman, K. (2016). River stage forecasting using wavelet packet decomposition and machine learning models. *Water Resources Management*, 30(11), 4011-4035.

Sethia, A., Raut, P. (2019). Application of LSTM, GRU and ICA for stock price prediction. In *Information and Communication Technology for Intelligent Systems* (pp. 479-487). Springer, Singapore.

Shafaei, M., Kisi, O. (2016). Lake level forecasting using wavelet-SVR, wavelet-ANFIS and wavelet-ARMA conjunction models. *Water Resources Management*, 30(1), 79-97.

Shewalkar, A., Nyavanandi, D., Ludwig, S.A. (2019). Performance evaluation of deep neural networks applied to speech recognition: RNN, LSTM and GRU. *Journal of Artificial Intelligence and Soft Computing Research*, 9(4), 235-245.

Shoaib, M., Shamseldin, A.Y., and Melville, B.W. (2014). “Comparative study of different wavelet based neural network models for rainfall-runoff modeling.” *Journal of Hydrology*, Vol. 515, pp. 47-58.

Shoaib, M., Shamseldin, A.Y., Melville, B.W., Khan, M.M. (2015). Runoff forecasting using hybrid wavelet gene expression programming (WGEP) approach. *Journal of Hydrology*, 527, 326-344.

Sivakumar, B., Berndtsson, R., Olsson, J., and Jinno, K. (2001). “Evidence of chaos in the rainfall- runoff process.” *Hydrological Sciences Journal*, Vol. 46, No. 1, pp. 131-145.

Solgi, A., Nourani, V., Pourhaghi, A. (2014). Forecasting Daily Precipitation Using Hybrid Model of Wavelet-Artificial Neural Network and Comparison with Adaptive Neurofuzzy Inference System (Case Study: Verayneh Station, Nahavand). *Advances in Civil Engineering*, 1-12, Doi10.1155/2014/279368.

Stone, M. (1974). “Cross-validatory choice and assessment of statistical predictions.” *Journal of the royal statistical society. Series B (Methodological)*, Vol. 36, No. 2, pp. 111-147.

Sudheer, K.P., Gosain, A.K., Ramasastri, K.S. (2002). A data-driven algorithm for constructing artificial neural network rainfall-runoff models. *Hydrological processes*, 16(6), 1325-1330.

Talei, A., Chye, C.L.H., and Wong, T.S.W. (2010). “Evaluation of rainfall and discharge inputs used by adaptive network-based Fuzzy Inference Systems (ANFIS) in rainfall-runoff modeling.” *Journal of Hydrology*, Vol. 391, pp. 248-262.

Takagi, T., & Sugeno, M. (1985). Fuzzy identification of systems and its applications to modeling and control. *IEEE Transactions on Systems, Man, and Cybernetics*, 15, 116–132.

Tarmoul, N., & Boulemtafes, B. (2018). Extreme rainfall and flooding in the watershed of the middle Sebaou, *Rev. Sci. Technol., Synthèse* 36: 39-48.

Tayyab, M., Zhou, J., Dong, X., Ahmad, I., Sun, N. (2017). Rainfall-runoff modeling at Jinsha River basin by integrated neural network with discrete wavelet transform. *Meteorology and Atmospheric Physics*, 1-11, Doi10.1007/s00703-017-0546-5.

Tiwari, M.K., Chatterjee, C. (2010). Development of an accurate and reliable hourly flood forecasting model using wavelet–bootstrap–ANN (WBANN) hybrid approach. *Journal of Hydrology*, 394, 458–470.

Torres, J.F., Troncoso, A., Koprinska, I., Wang, Z., Martínez-Álvarez, F. (2018). Deep learning for big data time series forecasting applied to solar power. In *The 13th International Conference on Soft Computing Models in Industrial and Environmental Applications* (pp. 123-133). Springer, Cham.

Uysal, G., Şorman, A.Ü. (2017). Monthly streamflow estimation using wavelet-artificial neural network model: A case study on Çamlidere dam basin, Turkey. *Procedia Computer Science*, 120, 237-244.

Wang, B., Sun, Y., Xue, B., Zhang, M. (2019). Evolving deep neural networks by multi-objective particle swarm optimization for image classification. In *Proceedings of the Genetic and Evolutionary Computation Conference*, 490-498.

Wang, W.C., Chau, K.W., Cheng, C.T., Qiu, L. (2009). A comparison of performance of several artificial intelligence methods for forecasting monthly discharge time series. *Journal of Hydrology*, 374, 294–306.

Wang, F., Wang, X., Chen, B., Zhao, Y., Yang, Z. (2013). Chlorophyll a simulation in a lake ecosystem using a model with wavelet analysis and artificial neural network. *Environmental management*, 51, 1044-1054.

Wang, W.Q., Goldnaraghi, M. F., Ismail, F. (2004). Prognosis of machine health condition using neuro-fuzzy Systems, *Mechanical Systems and Signal Processing*, 18813-831.

Wu, C.L., Chau, K.W., Fan, C. (2010). Prediction Of Rainfall Time Series Using Modular Artificial Neural Networks Coupled With Data-Preprocessing Techniques, *J. Hydrol*, 389, 146–167.

Yang, G., Lee, H., Lee, G. (2020). A hybrid deep learning model to forecast particulate matter concentration levels in Seoul, South Korea. *Atmosphere*, 11(4), 348.

Yu, H., Wen, X., Feng, Q., Deo, R.C., Si, J., Wu, M. (2018). Comparative study of hybrid-wavelet artificial intelligence models for monthly groundwater depth forecasting in extreme arid regions, Northwest China. *Water Resources Management*, 32(1), 301-323.

Yurdusev, M. A., & Firat, M. (2009). Adaptive neuro fuzzy inference system approach for municipal water consumption modeling: An application to Izmir, Turkey. *Journal of Hydrology*, 365, 225–234.

Zakhrouf, M., Bouchelkia, H., Stamboul, M. (2015). Neuro-Wavelet (WNN) and Neuro-Fuzzy (ANFIS) systems for modeling hydrological time series in arid areas. A case study: the catchment of Aïn Hadjadj (Algeria). *Desalination and Water Treatment*, 57, 17182-17194.

Zakhrouf, M., Bouchelkia, H., Stamboul, M., Kim, S. (2020). Novel hybrid approaches based on the evolutionary strategy for streamflow forecasting in the Chellif River, Algeria, *Acta Geophysica*, 68, 167-180.

Zakhrouf, M., Bouchelkia, H., Stamboul, M., Kim, S., Heddam, S. (2018). Time series forecasting of river flow using an integrated approach of wavelet multi-resolution analysis and evolutionary data-driven models. A case study: Sebaou River (Algeria), *Physical Geography*, 39(6), 506-522.

Zakhrouf, M., Bouchelkia, H., Stamboul, M., Kim, S., Singh, V.(2020 a). Implementation on the evolutionary machine learning approaches for streamflow forecasting: case study in the Seybous River, Algeria. *J. Korea Water Resour. Assoc.*, 53, 395-408.

Zakhrouf, M., Bouchelkia, H., Stamboul, M., Kim, S. (2020 b). Novel hybrid approaches based on the evolutionary strategy for streamflow forecasting in the Chellif River, Algeria, *Acta Geophysica*, 68, 167-180.

Zhang, J., Chung, H.S.H., Lo, W.L. (2008). Chaotic time series prediction using a neuro-fuzzy system with time-delay coordinates. *IEEE Transactions on Knowledge and Data Engineering*, 20, 956-964.

Zhao, X., Guo, X., Luo, J., and Tan, X. (2018). “Efficient detection method for foreign fibers in cotton.” *Information Processing in Agriculture*, Vol. 5, No. 3, pp. 320-328.

Zhu, S., Luo, X., Yuan, X., Xu, Z. (2020). An improved long short-term memory network for streamflow forecasting in the upper Yangtze River. *Stochastic Environmental Research and Risk Assessment*, 1-17.

Zuo, G., Luo, J., Wang, N., Lian, Y., He, X. (2020). Decomposition ensemble model based on variational mode decomposition and long short-term memory for streamflow forecasting. *Journal of Hydrology*, 124776.