Viharos, Zs. J.; Kis K. B.: Survey on Neuro-Fuzzy Systems and their Applications in Technical Diagnostics, *13th IMEKO TC10 Workshop on Technical Diagnostics, Advanced measurement tools in technical diagnostics for systems' reliability and safety*, June 26-27, 2014, Warsaw, Poland, ISBN 978-92-990073-3-4, pp. 87-92.

Survey on Neuro-Fuzzy Systems and their Applications in Technical Diagnostics

Dr. Zs. J. Viharos[1], K. B. Kis[1]

[1] *The Computer and Automation Research Institute, Hungarian Academy of Sciences, Budapest, H-1111 Hungary, {viharos.zsolt, kis.krisztian}@sztaki.mta.hu*

Abstract- Both fuzzy logic, as the basis of many inference systems, and Neural Networks, as a powerful computational model for classification and estimation, have been used in many application fields since their birth. These two techniques are somewhat supplementary to each other in a way that what one is lacking of the other can provide. This led to the creation of Neuro-Fuzzy systems which utilize fuzzy logic to construct a complex model by extending the capabilities of Artificial Neural Networks. Generally speaking all type of systems that integrate these two techniques can be called Neuro-Fuzzy systems. Key feature of these systems is that they use input-output patterns to adjust the fuzzy sets and rules inside the model. The paper reviews the principles of a Neuro-Fuzzy system and the key methods presented in this field, furthermore provides survey on their application for technical diagnostics. *Keywords*: Technical diagnostics, Neurofuzzy systems, Overview

I. Introduction

As two important techniques of artificial intelligence, fuzzy systems and Artificial Neural Networks (ANNs) have many applications in various fields such as production, control systems, diagnostic, supervision, etc. They evolved and improved throughout the years to adapt arising needs and technological advancements. As ANNs and Fuzzy Systems had been often applied together the concept of a fusion between them started to take shape. Neuro-Fuzzy Systems were born which utilize the advantages of both techniques: they have learning and generalization capabilities and at the same time they reveal the functionality stored in the model. To reach this behaviour they are able to learn and tune their parameters based on input-output patterns (learning phase) and then they work like a fuzzy logic system (execution phase), too. These combined features make this type of systems useful when solving complex problems also for technical diagnostic assignments.

The paper contains seven sections. After the introduction the second section presents Neuro-Fuzzy applications of the last two decades in technical diagnostics. The third section describes the two main components of a Neuro-Fuzzy system followed by the forth one reviewing the progression of the Neuro-Fuzzy systems and the modern solutions used today. The last three sections are conclusions, acknowledgments and references.

II. Application of Neuro-Fuzzy Systems to Technical Diagnostics

This section gives a survey on Neuro-Fuzzy system applications in the field of technical diagnostics. Different Neuro-Fuzzy architectures are named here and their history and a more detailed description are presented in the next sections.

In the early 90s Neuro-Fuzzy was still a new concept to be shaped by different implementations and applications. For example Ayoubi presented a structure that models the fuzzy inference mechanism based on neural units [1]. He tested the system on two real-world problems: monitoring the state of a turbocharger and supervision of air pressure in vehicle wheels. The structure proved to be efficient when the problem space was small; however the conclusion was that, in higher dimensions, Multi-Layer Perceptron (MLP) can perform far better than the fuzzy inference mechanism. Zhang and Morris also used a Neuro-Fuzzy solution for fault diagnosis of continuous stirred tank reactor process [2]. Their network consists of 4 layers: an input layer, a fuzzification layer, a hidden layer and an output layer. The input layer has 14 neurons, the fuzzification layer has 3 neurons for each input neuron, the hidden layer has 10 neurons and the output layer has 11 neurons, each corresponding to a particular fault. They achieved much better performance than with a conventional feed forward neural network while the system also provided a more interpretable structure.

Neuro-Fuzzy systems became more widespread in the 2000s especially in technical diagnostics. Detecting the onset of damage in gear systems was the goal of Wang et al., for which they developed a neuro-fuzzy based diagnostic system [3]. The diagnosis of the gear system is conducted gear-by-gear, which means that for every gear there is a separated neuro-fuzzy model. Each model has three inputs and one output: the inputs are reference functions that reduce the feature dimensions, i.e. they aggregate multiple features of the real system to one index; the output is the condition of the gear, which can be normal or damaged. To train the implemented model they proposed a constrained-gradient-reliability algorithm which can effectively update the membership function parameters and set the rule weights.

Evsukoff and Gentil created a recurrent Neuro-Fuzzy system for fault detection and isolation in nuclear reactors [4]. In their model a fuzzification module is linked to a neural network based inference module which was adapted to recognize related faults based on the process variables.
One of the first and probably most widespread Neuro-Fuzzy architecture is the ANFIS which has similar accuracy as the MLP which makes it ideal for function approximation. This architecture was used for mechanical fault diagnostics of induction motors with variable speed drives by Sadeghian and Wu [5]. The authors managed to significantly reduce the system complexity and learning duration of the network by using multiple ANFIS units in their model.

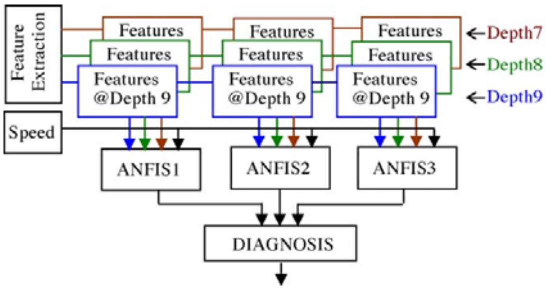

Figure 1. Multiple ANFIS units for multiple fault diagnostics [5]

Figure 1. shows the multiple ANFIS units where each one is responsible for detecting a specific fault type as these fault types have different feature coefficients. This modular structure provides an easy way to make extensions for detecting other fault types and also has the advantage that the units can be easily trained due to their simplicity. In another application Lei et al. used multiple ANFIS combination with genetic algorithm for fault diagnostics of rotating machinery [6]. They implemented a classifier system where the features describing the problem were divided into six groups and individual neuro-fuzzy classifiers were constructed for each group. The final classification result of the system is the weighted average of the individuals where genetic algorithm is used for optimising the weights. This method can yield better classification result than the member classifiers individually.
Machinery malfunctions often reduce productivity and increase maintenance costs in various industrial fields. Zio and Gole proposed a neuro-fuzzy approach to solve fault diagnostic problems by pattern classification while obtaining a model which remained easily interpretable [7]. Their algorithm consists of multiple modules. First, an initial set of fuzzy rules are determined, where the initial large number of rules is reduced with a heuristic solution based on the firing strength of each rule. Then the forward algorithm calculates the relative strengths of the rules and the next module uses these values for creating new rules if necessary. The optimisation module tunes the parameters of the member functions and finally, a pruning is applied to reduce the size of the rule set. After the initial set of rules has been established, the algorithm repeats itself iteratively until the desired accuracy is reached.
Chen, Roberts and Weston used Neuro-Fuzzy Systems for fault detection and diagnostics of railway track circuits [8]. In their solution, they use a generalized version of the ANFIS to support multiple outputs. Each one of the eleven outputs corresponds to a condition (1 healthy and 10 faulty condition) while eight current and voltage measurements are used as the input variables.
Different application fields are also targeted by Neuro-Fuzzy solutions as in the case of another ANFIS model which was used to detect alterations in sleep EEG activity during hypopnoea episodes by Übeyli et al. [9]. The authors used the ANFIS for classification and they performed feature extraction by computing of wavelet coefficients. In their case four models was used: three were fed directly by measured data on the electrodes and the fourth had the purpose of improving diagnostic accuracy by gaining its inputs from the outputs of the other three systems. There is a wide variety of other applications where this kind of systems was successfully implemented from the fields of biology and environment to fault detection and diagnostics as by Kar et al. [10].
Many other applications in technical diagnostics use different systems and methods for fault detection and other diagnostic purposes. Bilski used an artificial intelligence-based model for diagnostics of analog systems [11]. He preprocessed the training and testing data sets using statistical methods to minimize the amount of information to be measured in the actual system. In another application Catelani and Ciani analysed the problem of disturbance induced by high energy particles on electronic devices and developed a model to determine whether a system respond to specific requirements [12].

III. Prelude of Neuro-Fuzzy Systems

This section discusses the techniques that provided the theoretical basis and allowed the concept of Neuro-Fuzzy system to be formed. These techniques are the Neural Networks and the Fuzzy Systems which will be presented in the following paragraphs.

A. Artificial Neural Networks

As the name suggests Artificial Neural Networks (ANNs) were created based on the behaviour of biological neural networks inside the human brain. ANN is a structure where autonomous computational units (neurons) are connected via weighted links. A general neuron [13] consists of input and output links (special neurons sometimes don't have any input or output links), a transfer function, an activation function and an optional memory component. The transfer function processes the outputs of the other neurons connected via the input links then the activation function produces the output value and the memory serves as a container to store previous states of the neuron (in many cases it is not used or only a part of the state is stored). When a neuron fires the signal propagates through the output links to the connected neurons and the weight of a link (represented by a real number) determines the strength of the connection and weakens the signal accordingly. The training of an ANN is to adjust the weight of each link to receive more and more favourable output on specific neurons (output neurons) while stimulating other neurons (input neurons).
The most widespread structure is organizing the neurons into layers where each neuron of a layer is connected to every neuron of the next layer and to those only. This structure is called Multi-Layer Perceptron (MLP) [14]. During the training of an MLP input-output patterns are used to reduce the error of the output neurons, e.g. the weights are adjusted during the process in such a way that the output values of the network come more and more close to the ones in the patterns for the same input.
ANNs are powerful computational models for solving complex estimation and classification problems as they are robust and are capable of high level generalization, moreover they can already handle incomplete data, too [15]. However no information can be extracted from a trained ANN about the connections between the parameters, e.g. a generic ANN model can only approximate the output parameters but cannot tell what kind of connections exist between the input and output parameters. This is a key disadvantage of the Neural Network model which led to the creation of Neuro-Fuzzy Systems.

B. Fuzzy Systems

Fuzzy logic provides an effective way to represent human knowledge in a mathematical language. The fuzzy sets were introduced by Lofti Zadeh [16] where the behaviour of the system is described by fuzzy rules. The behaviour of such systems is described through a set of fuzzy rules, like: *if <premise> then <consequent>* that uses linguistics variables with symbolic terms. Each term represents a fuzzy set. The terms of the input space (typically 5-7 for each linguistic variable) compose the fuzzy partition. The fuzzy inference mechanism consists of three stages: in the first stage, the values of the numerical inputs are mapped by a function according to a degree of compatibility of the respective fuzzy sets; this operation can be called fuzzyfication. In the second stage, the fuzzy system processes the rules in accordance with the firing strengths of the inputs. In the third stage, the resultant fuzzy values are transformed again into numerical values; this operation can be called defuzzyfication. Essentially, this procedure makes possible the use fuzzy categories in representation of words and abstracts ideas of the human beings in the description of the decision taking procedure [17].
Fuzzy Systems have the advantage that the fuzzy rules, which store the information, are easily interpretable. Furthermore they provide a simple interface for extending the system with new information (by adding new rules) or manipulating the existing rules. The problem with Fuzzy Systems lies in the fact that they completely depend on the experts who design them. It only uses the information which were encoded in the system and cannot learn on its own and incapable of generalization. The described nature of Fuzzy Systems indicates that a fusion with ANNs may possibly lead to a new powerful computational model.

IV. Neuro-Fuzzy System Architectures

The previous section briefly described the concept of the two main components building up a Neuro-Fuzzy system individually, so in this section the different architectures can be discussed to show how different approaches managed to combine ANNs with Fuzzy Systems.

A. ANFIS Architecture

One of the first Neuro-Fuzzy Systems was introduced by Jang [18][19]. This architecture is called ANFIS (Adaptive-Network-based Fuzzy Inference System) and it uses the Takagi-Sugeno inference system.
Figure 2. shows the ANFIS architecture which consists of six layers. The first layer contains two nodes for input

x and y, the second layer is responsible for mapping input values to the membership functions. The nodes of the third layer correspond to the fuzzy rules in the form of production functions; their output values are the firing strengths of each rule while the nodes in the fourth layer calculate the ratio to the sum of all rules' firing strengths. Defuzzyfication happens in the fifth layer and the sixth layer's output nodes sum their input values.
Iterative learning of ANFIS is composed of two stages. In the first stage the parameters of the consequent functions (in the fifth layer) are tuned via a least mean square method. During the second stage the parameters of the premise functions (in the second layer) are adjusted by a backpropagation algorithm. These two stages are repeated iteratively for training of the system.

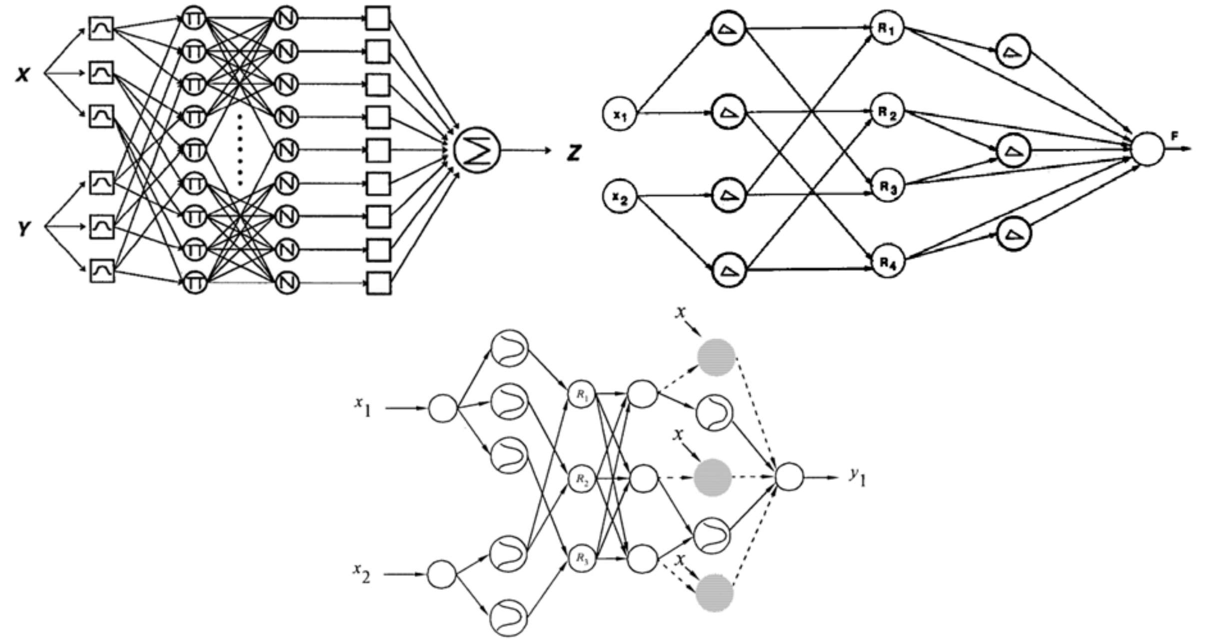

Figure 2. the ANFIS [18] (top left), the GARIC ASN [21] (top right) and the SONFIN [25] (bottom) architecture

B. GARIC Architecture

The GARIC (Generalized Approximate Reasoning based Intelligence Control) system is composed of three components: the ASN (Action Selection Network), the AEN (Action Evaluation Network) and the SAM (Stohastic Action Modifier) [21]. Figure 2. shows the ASN component. The ASN is a five layer network which is responsible for selecting an action based on the current state of the system using fuzzy inference. Input nodes are in the first layer and the second one holds the membership functions. Each node in the third layer represents a fuzzy rule and nodes of the forth layer correspond to consequent labels, e.g. if a consequent label is in a rule then there is a link between the label's node and the rule's node. The fifth layer's nodes calculate the real output values based on the rules' firing strength and the forth layer's outputs. AEN is a simple feedforward network which predicts reinforcements based on the state variables of the system. GARIC uses gradient descending and reinforcement learning to adjust its internal parameters.

C. SONFIN Architecture

SONFIN (Self-Constructing Neural Fuzzy Inference Network) is a Takagi-Sugeno-Kang-type fuzzy rule-based model which consists of six layers [25]. Figure 2. shows the SONFIN architecture which, in fact, is similar to the ANFIS. Layer 1-4 and 6 are functioning as they are in the ANFIS architecture. The fifth, consequent layer can hold two types of nodes. The first type represents the fuzzy sets by membership functions while the second type is optional and gains its inputs from the first and fourth layer.
Constructing of SONFIN happens concurrently by a structure and a parameter learning method. The structure learning identifies both the precondition and consequent parts of the rules by minimizing the number of rules and membership functions for the input and by optimally generating new membership functions for the output variables. Parameter learning uses LMS or RLS algorithms to adjust consequent parameters and backpropagation for precondition parameters.
The following sections describe other neuro-fuzzy architectures without visual presentation of their structure.

D. FALCON Architecture

FALCON (Fuzzy Adaptive Learning Control Network) is a system with five layers [20]. Input nodes are located in the first layer; second layer has term nodes which represent the membership functions for the input values. Each node of the third layer acts as a fuzzy rule. The fourth layer also consists of term nodes; these represent the membership functions for the outputs. Finally the fifth layer is the output layer; here for every output there are two nodes: one is for training data which is the desired output and the other is for decision signal which is the actual output.

Training is done by a two-phase-algorithm. The first phase is responsible for finding the initial membership functions by a self-organized learning scheme. In the second phase the parameters of the membership functions are adjusted using supervised learning. During the training nodes and links can be deleted or combined reforming the structure of the network.

E. NEFCON Architecture

The NEFCON (Neural Fuzzy Controller) has three layers and implements a Mamdani type inference system [22]. The first layer consists of the input nodes, in the second layer the nodes represent the fuzzy rules and the third layer holds the output nodes. In this architecture the links connecting the nodes are weighted with fuzzy sets.

The learning procedure uses reinforcement learning with backpropagation algorithm to either learn the rule base from the beginning or to optimise an initially defined rule base. Two other systems were developed based on NEFCON which are specialized versions of the original architecture. These systems are the NEFCLASS [23] which is specialized in classification problems and the NEFPROX [24] which was created for function approximation.

F. dmEfuNN Architecture

The dmEfuNN (Dynamic Evolving Fuzzy Neural Network) is a system with five layers which uses the Takagi-Sugeno fuzzy inference [26]. The predecessors of this model are FuNN [27] and EFuNN [28] which both uses the Mamdani type inference. The first layer contains the input nodes and membership functions are in the second layer. Fuzzy rules are represented by the nodes in the third layer. Fourth layer selects a number of rules from the third layer which are the closest to the fuzzy inputs and the fifth layer does the defuzzycation and produces real outputs.

The dmEfuNN can optimize global generalization error and local generalization error in contrast to MLP and ANFIS which can only optimize global error. As the name suggests the number of nodes and links in the structure can dynamically increase or decrease during the on-line learning while off-line training uses a given structure and optimizes internal parameters.

V. Conclusions

Different applications of Neuro-Fuzzy Systems were discussed to show their high potential in technical diagnostics. These systems are successful because of their nature that they reveal the nature of the important interdependence between the parameters of the modelled system while they are, in fact, powerful approximators.

The paper briefly reviewed the concept of Artificial Neural Networks and Fuzzy Systems as computational models and how they inspired the creation of Neuro-Fuzzy Systems. As it was discussed this fusion can unite the generalization capabilities of Neural Networks with the easy interpretability and high expressive power of fuzzy rules in an effective way.

Six different architectures were presented and it can be concluded that these are the most important ones although there are other structure variations, too. Usually each architecture organizes its nodes a slightly different way and consequently they use specific learning algorithms which are adapted to the different structures.

VI. Acknowledgments

The authors acknowledge the support of grants of the Fraunhofer Project Center for Production Management and Informatics at SZTAKI, Budapest, Hungary and the Highly industrialised region on the west part of Hungary with limited R&D capacity: Research and development programs related to strengthening the strategic future-oriented industries manufacturing technologies and products of regional competences carried out in comprehensive collaboration, VKSZ_12-1-2013-0038.

References

[1] M. Ayoubi, “Fuzzy systems design based on a hybrid neural structure and application to the fault diagnosis of technical processes”, *Control Engineering Practice*, vol. 4: (1), pp. 35-42, 1996.
[2] J. Zhang, J. Morris, “Process modelling and fault diagnosis using fuzzy neural networks”, *Fuzzy Sets and Systems*, vol. 79, pp. 127-140, 1996.
[3] W. Wang, F. Ismail, A. F. Golnaraghi, “A neuro-fuzzy approach to gear system monitoring”, *IEEE Transactions on Fuzzy Systems*, vol. 12: (5), pp. 710-723, 2004.
[4] A. Evsukoff, S. Gentil, “Recurrent neuro-fuzzy system for fault detection and isolation in nuclear reactors”, *Advanced Engineering Informatics*, vol. 19, pp. 55-66, 2005.
[5] Z. Ye, A. Sadeghian, B. Wu, “Mechanical fault diagnostics for induction motor with variable speed drives using Adaptive Neuro-fuzzy Inference System”, *Electric power System Research*, vol. 76, pp. 742-752, 2006.
[6] Y. Lei, Z. He, Y. Zi, Q. Hu, “Fault diagnosis of rotating machinery based on multiple ANFIS combination with GAs”, *Mechanical Systems and Signal Processing*, vol. 21: (5), pp. 2280-2294, 2007.
[7] E. Zio, G. Gola, “A neuro-fuzzy technique for fault diagnosis and its application to rotating machinery”, *Reliability Engineering and System Safety*, vol. 94, pp. 78-88, 2009.
[8] J. Chen, C. Roberts, P. Weston, “Fault detection and diagnosis for railway track circuits using neuro-fuzzy systems”, *Control Engineering Practice*, vol. 16: (5), pp. 585-596, 2008.
[9] E. D. Übeyli, D. Cvetkovic, G. Holland, I. Cosic, “Adaptive neuro-fuzzy inference system employing wavelet coefficients for detection of alterations in sleep EEG activity during hypopnoea episodes”, *Digital Signal Processing*, vol. 20, pp. 678-691, 2010.
[10] S. Kar, S. Das, P. K. Ghosh, “Applications of neuro fuzzy systems: A brief review and future outline”, *Applied Soft Computing*, vol. 15, pp. 243-259, 2014.
[11] P. Bilski, “Data set processing for the optimization of the artificial intelligence-based diagnostic methods”, IMEKO TC10 Workshop on Technical Diagnostics, Florence, Italy, 2013.
[12] M. Catelani, L. Ciani, “Diagnostic and Error Correction System for Avionics Devices in Presence of Single Event Upset (SEU)”, IMEKO TC10 Workshop on Technical Diagnostics, Florence, Italy, 2013.
[13] W. S. McCulloch, W. Pitts, “A logical calculus of the ideas immanent in nervous activity”, *Bulletin of Mathematical Biophysics*, vol. 5, pp. 115-133, 1943.
[14] P. J. Werbos, *Beyond Regression: New Tools for Prediction and Analysis in the Behaviour Sciences*, Ph. D Thesis, Harward University, Cambridge, 1974.
[15] Zs. J. Viharos, K. B. Kis, “Diagnostics of wind turbines based on incomplete sensor data”, IMEKO World Congress Technical Diagnostics, Republic of Korea, 2012.
[16] L. A. Zadeh, “Fuzzy Sets”, *Information and Control*, vol. 8, pp. 338-353, 1965.
[17] J. Vieira, F. M. Dias, A. Mota, “Neuro-Fuzzy Systems: A Survey”, 5th WSEAS NNA International Conference, 2004.
[18] R. Jang, “Fuzzy Modeling Using Generalized Neural Networks and Kalman Filter Algorithm”, *Proc. Ninth Nat. Conf. Artificial Intell.*, pp. 762-767, 1991.
[19] R. Jang, “ANFIS: Adaptive-Network-Based Fuzzy Inference System”, *IEEE Transactions on systems, man, and cybernetics*, vol. 23: (3), pp. 665-685, 1993.
[20] T. C. Lin, C. S. Lee, “Neural Network Based Fuzzy Logic Control and Decision System”, *IEEE Transactions on Computers*, vol. 40: (12), pp. 1320-1336, 1991.
[21] H. R. Berenji, P. Khedkar, “Learning and Tuning Fuzzy Logic Controllers Through Reinforcements”, *IEEE Transactions on Neural Networks*, vol. 3, pp. 724-740, 1992.
[22] D. Nauck, R. Kruse, “NEFCON-I: An X-Window Based Simulator for Neural Fuzzy Controllers”, *Proc. IEEE Int. Conf. Neural Networks*, 1994.
[23] D. Nauck, R. Kruse, “NEFCLASS – A Neuro-Fuzzy Approach For The Classification Of Data”, *Applied Computing Proc. of the 1995 ACM Symposium on Applied Computing*, pp. 461-465, 1995.
[24] D. Nuck, R. Kurse, “Neuro-fuzzy systems for function approximation”, *Fuzzy Sets and Systems*, vol. 101, pp. 261-271, 1999.
[25] F. C. Juang, T. C. Lin, “An On-Line Self Constructing Neural Fuzzy Inference Network and its applications”, *IEEE Transactions on Fuzzy Systems*, vol. 6, pp. 12-32, 1998.
[26] N. Kasabov, Q. Song, *Dynamic Evolving Fuzzy Neural Networks with ‘m-out-of-n’ Activation Nodes for On-Line Adaptive Systems*, Technical Report TR99/04, 1999.
[27] N. Kasabov, “Adaptable Connectionist Production Systems”, *Neurocomputing*, vol. 13: (2-4), pp. 95-117, 1996.
[28] N. Kasabov, “Evolving Fuzzy Neural Networks – Algorithms, Applications and Biological Motivation”, *Methodologies for the Conception, Design and Application of Soft Computing*, pp. 271-274, 1998.