

# A Fuzzy Rough Rule Based System Enhanced By Fuzzy Cellular Automata

Mona Gamal<sup>1</sup>, Ahmed Abou El-Fetouh<sup>2</sup>, Shereef Barakat<sup>3</sup>  
Mansoura University, Faculty of Computer and Information Sciences  
Information System Department  
P.O.Box: 35516

**Abstract**—Handling uncertain knowledge is a very tricky problem in the current world as the data, we deal with, is uncertain, incomplete and even inconsistent. Finding an efficient intelligent framework for this kind of knowledge is a challenging task. The knowledge based framework can be represented by a rule based system that depends on a set of rules which deal with uncertainty in the data. Fuzzy rough rules are a good competitive in dealing with the uncertain cases. They are consisted of fuzzy rough variables in both the propositions and consequences. The fuzzy rough variables represent the lower and upper approximations of the subsets of a fuzzy variable. These fuzzy variables use labels (fuzzy subsets) instead of values. An efficient fuzzy rough rule based system must depend on good and accurate rules. This system needs to be enhanced to view the future recommendations or in other words the system in time sequence. This paper tries to make a rule based system for uncertain knowledge using fuzzy rough theory to generate the desired accurate rules and then use fuzzy cellular automata parallel system to enhance the rule based system developed and find out what the system would look like in time sequence so as to give good recommendations about the system in the future. The proposed model along with experimental results and simulations of the rule based systems of different data sets in time sequence is illustrated.

**Keywords**—fuzzy rough reduction; fuzzy rough rules; fuzzy cellular automata; Self Organized Feature Maps (SOFM).

## I. INTRODUCTION

Knowledge Discovery (KD) [25] is the process of extracting valuable knowledge from concrete data sets. This process used to be accomplished manually or semi manually (part manual and part automated). The aim of soft computing techniques[15] is to completely automate the process of KD. The problem is that the data we want to extract knowledge from is uncertain, incomplete and imprecise. So we need to represent vague concepts of information. Rule based systems are composed of a set of if-then rules that represent the knowledge content of the system.

Many soft computing techniques were used for various problems in the KD process. For example, Genetic Algorithms (GAs) [5] [8] [27] [32] were used for optimization and search problems. Rough sets[31] and Artificial Neural Networks[18] are very good tools for classification and prediction problems. They are efficient in dealing with discrete data, however the real world is dealing with values like tall, short, normal, up normal and so on. Fuzzy set theory[19] [20] which deal with linguistic values of the variables are to be used here to create

rules which handle the linguistic world's problems. The hybridization between fuzzy system and various soft computing techniques is a very interesting search topic these days. Hybridizations like fuzzy rough, fuzzy neural, fuzzy genetic algorithms and many others are very powerful in dealing with uncertain knowledge in linguistic form away from the complicated mathematical calculations of probabilities. The fuzzy rough hybrid[28] is very interesting in the field of building equivalence classes with soft boundaries and degrees of membership of the objects inside these classes.

Another interesting mechanism is the fuzzy cellular automata [11] . It is a parallel processing system that is composed of a set of interconnected cells. These systems can be used efficiently to build a grid of rules in time sequence based on initial if-then rules produced from a soft computing rule generating technique. This could be very helpful for experts on the field of the data set under consideration as they need to see what the system would look like in the future. So the cellular automata can be used to enhance the rule based system to reduce the error rate or examine in which variable direction the knowledge discovered goes.

This research is concerned in producing a complete framework for KD by building a rule based system and then enhances it with the fuzzy cellular automata. This process is accomplished in three phases of a hybrid system which type is a transformation system (the output of one module is an input to the preceding module). The first module is to prepare the fuzzy variables by generating the membership function for the fuzzy subsets of the variables. This module is implemented using Self Organized Feature Maps (SOFM) [29] from a previous research. The result of this phase is passed to the fuzzy rough rule generating module which reduces the attributes first to get the reduct (the attributes which the data set fully depend on them with no redundancies) then this reduct is used to summarize the fuzzy data and produce the corresponding fuzzy rough rules. These rules are tested against the test data to measure the accuracy rates. The set of fuzzy rough rules are the core of the rule based system that represents the data set. The final module takes the rules as an initial state for the fuzzy cellular automata parallel system that iterate to generate new rules to cover the whole corresponding data space with the suitable rules that represent new data objects. This could be thought of as an enhancement of the system equation (the set of fuzzy rough rules) or the recommendations of the system in the time sequence.

The rest of this paper is organized as follows: Section II is a quick review on the previous research in generating fuzzy rules and using cellular automata in the data mining field. Section III represents the preliminaries and theories such as the declaration of the Self Organized Feature Maps (SOFM), Fuzzy rough attribute reduction, rule generation and cellular automata. Section IV gives an over view on the whole system and its modules. It goes inside the system to explain in detail the generation of the fuzzy membership functions of the subsets of the fuzzy variables, attribute reduction, designing the fuzzy rough rule set and enhancing the system using cellular automata parallel system. Experimental results and conclusion will appear in sections V and VI respectively.

## II. RELATED WORK

Designing a complete framework in the KD field is a very interesting topic that many researchers cared about and tried to find the best system to accomplish the job. The system contains problems such as preparing the attributes, finding the reduct and generating the rule set that represent the core of the system. The system produced will certainly need an enhancement to its accuracy and the scope it covers on the data space. These problems many researches covered it individually in a hope to find good and efficient solutions. The reducing attribute problem is important to remove redundancy. Rough set theory is a very good way to get the reduct from crisp data but the fuzzy data may have some loss of information. A new dimensionality reduction technique that employs a hybrid variant of rough sets, fuzzy-rough sets, to avoid this information loss[30]. Genetic Algorithm[5] [8] [27] [32] [27] has been applied for the discovery of fuzzy rules which were competitive to decision tree induction rules from the perspective of predictive accuracy[32]. The individuals of the population were the fuzzy rules to be designed and the final population was the fuzzy rule set. A hybrid algorithm of two fuzzy genetics-based machine learning approaches (i.e., Michigan and Pittsburgh) for designing fuzzy rule-based classification systems has also been proposed[12] [13] [14]. A new method was also proposed to automatically learn the knowledge base (KB) by finding an appropriate database by means of a genetic algorithm while using a simple generation method to derive the rule base (RB) [24]. Also a new hybrid approach for optimization combining Particle Swarm Optimization (PSO) and Genetic Algorithms (GAs) using Fuzzy Logic for parameter adaptation and to integrate the results. Fuzzy Logic is used to combine the results of the PSO and GA in the best way possible. Also, fuzzy logic is used to adjust parameters in the FPSO and FGA[7] [8]. Other research described the use of Modular Neural Networks (MNN) for pattern recognition in parallel using a cluster of computers with a master-slave topology. Also, a parallel genetic algorithm to optimization architecture was used Fig. 8. But these researches assume that fuzzy variables are prepared or use discrimination techniques to make the membership functions of the subsets of the variables. So some researches thought of a technique for estimating the fuzzy subsets. Gene Expression Programming method uses two populations. One for Fuzzy Classification Rules which is evolved by syntax genetic programming and the other one for membership function definitions which is evolved by mutation based

evolutionary algorithm. These two populations co-evolve to better classify the underlying data set[1] [24]. A rough fuzzy hybridization to generate fuzzy if-then rules automatically from diagnoses data sets with quantitative data values based on fuzzy sets and rough set theory is accomplished in four different stages in the KD from databases[23].

There are many papers that introduced fuzzy cellular automata parallel system in the KD field to make use of its simplicity and efficiency. A fuzzy cellular traffic model is proposed which intended for detectors data fusion in traffic control system to enable utilization of complex traffic data registered by many sensors of different type's[3]. An other research introduced a cellular automata-based solution of a two-dimensional binary classification problem[2]. The proposed method is based on a two-dimensional, three-state cellular automaton (CA) with the von Neumann neighborhood. Since the number of possible CA rules (potential CA-based classifiers) is huge, searching efficient rules is conducted with use of a genetic algorithm (GA). An other paper shows a definition of a fuzzy automaton, which has the state, input, and output sets as fuzzy sets. The state transition function is defined as moving on a fuzzy relief with fuzzy peak-states and boundaries between different membership functions[17].

## III. PRELIMINARIES AND THEORIES

### A. Self Organized Feature Map

The Self Organized Feature Map (SOFM) [4] [29] is an unsupervised neural network that is capable of learning its weights from its input vector without supplying it with the corresponding output vector. SOFM is called self organizing or self adoption because they are able to decide what features it will use to group the input data. SOFM is usually, a two-layered network where the neurons in the output layer are organized into either a one or two-dimensional lattice structure (Bose and Liang, 1996). The SOM solves difficult high-dimensional and nonlinear problems such as feature extraction and classification of images and acoustic patterns, adaptive control of robots, and equalization, demodulation, and error-tolerant transmission of signals in telecommunications[29] and in this research it is used to find the membership function for fuzzy variables subsets. Figure 1 represents a simple structure for the SOFM where the dimension  $d$  is the number of the input neurons in the input layer for the following input data vector  $x_n = [x_{n1} \ x_{n2} \ \dots \ x_{nd}]^T$  and The synaptic weight vector at neuron  $j$  in the output layer is denoted by  $w_j = [w_{j1} \ w_{j2} \ \dots \ w_{jd}]^T$ ,  $j = 1, 2, \dots, J$ , where  $J$  is the total number of neurons in the output layer and  $w_{jk}$ ,  $k = 1, 2, \dots, d$ , is the connecting weight from the  $j^{\text{th}}$  neuron in the output layer to the  $k^{\text{th}}$  neuron in the input layer.

In the learning phase, the first step is to find the best matching neuron in the output layer that is the closest to the input vector from the following equation:

$$q(x_n) = \min_{\forall j} \|x_n - w_j\| \quad (1)$$

Where,

$q(x_n)$  is the index of the winning neuron in the output layer,

$x_n$  is the input vector,

$w_j$  is the weight vector between the input vector and the output neuron  $j$ ,

$\| \cdot \|$  is a distance measure (usually the Euclidean norm).

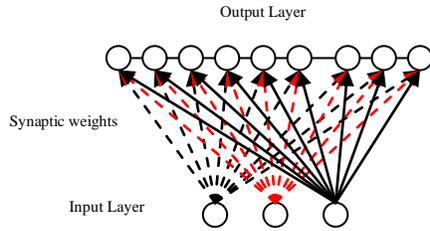


Fig. 1. a simple structure for the SOFM

The next step is to update the weight vectors associated with the winning neuron  $q$  ( $x_n$ ). The learning rule for neuron  $j \in N_q$  where  $N_q$  is the chosen neighborhood of winning neuron  $q$  for input vector  $x_n$ , is given by

$$w_j[t+1] = w_j[t] + \eta_{qj}[t](x_n[t] - w_j[t]) \quad (2)$$

Where

$$\eta_{qj}[t] = \begin{cases} \mu[t] & j \in N_q \\ 0 & j \notin N_q \end{cases} \quad (3)$$

Here,  $\mu[t]$  is the learning rate,  $0 < \mu[t] < 1$ , at time index  $t$ .

In the retrieving phase, when  $x_n$  is the input vector, only the winning neuron, after convergence, will have positive response.

### B. Fuzzy rough sets

Fuzzy rough set[28] is a generalization of the lower and upper approximation of the rough set[31] to allow soft boundaries. The thinking has been changed from objects which are indistinguishable (according to their attribute values) to objects similarity.

Objects are divided into fuzzy equivalence classes according to their similarities where an object could belong to more than one class with different degrees of membership. All equivalence classes are fuzzy this means that the decision values and the conditional values are all fuzzy. The lower and upper approximations are now:

$$\mu_{\underline{X}}(F_i) = \inf_x \max\{1 - \mu_{F_i}(x), \mu_X(x)\} \quad \forall i \quad (4)$$

$$\mu_{\overline{X}}(F_i) = \sup_x \min\{\mu_{F_i}(x), \mu_X(x)\} \quad \forall i \quad (5)$$

where  $F_i$  denotes a single fuzzy equivalence class. The tuple  $\langle \underline{X}, \overline{X} \rangle$  is called a fuzzy-rough set. Again, it can be seen that these definitions degenerate to traditional rough sets when all equivalence classes are crisp. Additionally, if all  $F_i$  s are crisp, the result is a rough set.

### C. Fuzzy rough attribute reduction

The non relevant attributes in data sets make the classification process more complicated as they take processing time and space and do not make any improvement in the classification accuracy level. The attribute reduction process aims to find those non relevant attribute and remove them from data sets before getting conducted in the classification process. The problem here is how to find those attribute carefully without affecting the final classification result. The measure of dependency between the attributes and the overall data set is reliable. For the uncertain knowledge and fuzziness issues, the fuzzy rough attribute reduction (FRAR) is an efficient and previously tested algorithm. The FRAR Algorithm is a new dimensionality reduction technique that employs a hybrid variant of rough sets (fuzzy-rough sets) in calculating the dependency between attributes and the data set avoiding information loss[30]. FRAR make use of the concepts of vagueness (for fuzzy sets) and indiscernibility (for rough sets) to measure the dependency degree between the fuzzy attributes and the uncertain data sets using the membership function of each attribute and the membership degree of the objects in each class to find the best reduct. The dependency degree is calculated over the attribute fuzzy positive region. The attribute that maximizes the overall dependency is added to the core attribute sets. The equations are illustrated in section IV.B.

### D. Fuzzy rough rule generation

The fuzzy rough rules in a rule based system are if-then rules with fuzzy rough attributes in the conditional and the consequences parts. These if-then rules can be easily found by summarizing the reduced data set resulted from the attribute reduction process. This method in summarizing the data set is used previously in a research paper for rule extraction using soft computing techniques[23]. The problem in fuzzy data is that there are real values with membership degrees that may be a difficulty for the rough set technique to discretize such data and get the if-then rules. The technique here is to transform the attribute real value into a tuple that represent the membership degrees in all the attribute subsets.

For example the fuzzy attribute height is represented by the membership function illustrated in figure 2. The value 170 can be written in the format (S: 0.0; M: 0.3; T: 0.7) where S, M and T are the attribute subsets and 0.0, 0.3, 0.7 are the membership degrees in each subset respectively. In the data summarizing process, we try to type this tuple in the 0's and 1's format so the rough set theory can be used to summarize the core data set and find all the if-then rules that represent the core data set. The 1's are given to the membership degrees more than or equal to 0.5 and 0's otherwise. The previous tuple can be written as (0; 0; 1). If there are two membership degrees in the tuple of value 0.5 then both of them will be 1 (i.e. (L:0.5;M:0.5;H:0.0) will be (1;1;0)).

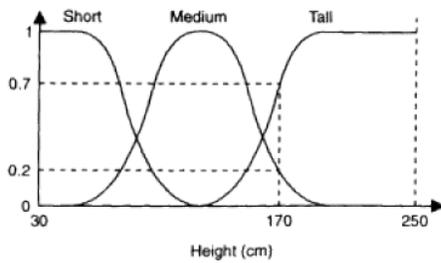


Fig.2. Membership functions representing three fuzzy sets for the variable "height."

E. Cellular automata parallel system

Cellular automata(CA)[16] is a parallel distributed processing system that aims to build a grid of cells from some initial configuration in the time sequence iterations according to some transition function (update rule). In 1d grids, there is a vector of cells where each cell has a state. The cell changes its state according to the states of its neighborhood's states and its state in the previous time step. Following [Wolfram, 1984], [Wolfram, 2002] one can represents any CA with two parameters (k; r).

Where k is a number of states, and r is a radius of neighborhood. Thus CA is defined by parameters (2; 1). There are  $n = k2r+1$  different neighborhoods and  $kn$  different evolution rules. For a cellular automata vector where each cell has 2 states and 1 radius tall then the number of rules that design the cell states in the next time step will be 265 rules. These rules are called transition functions that can be written in the form

$$C_i(t+1) = F(\dots, c_{i-1}(t), c_i(t), c_{i+1}(t+1), \dots) \quad (6)$$

Where

$c_i$  is the cell state,

$\dots, c_{i-1}(t), c_i(t), c_{i+1}(t+1), \dots$  are the cell neighbors,

F is the transition function.

In 2d and 3d cellular grids become very large, resulting in time and space consuming problems, so researchers have made some rules to generate the cells in the next time step without the conventional updating rules. It is called game of life, as it simulates human generation and existence, where the neighbors of the cells determine the cell state in the next time step. Von Neumann neighborhood and Moore neighborhood are two common neighborhood definitions for the cells in a two dimension cellular grid.

In von Neumann, each cell has neighbors to the north, south, east and west. The Moore neighborhood adds the diagonal cells to the northeast, southeast, southwest and northwest. Figure 3 shows these two neighborhood models in two dimensions. In general, in a d-dimensional space, a cell's von Neumann neighborhood will contain  $(2*d)$  cells and its Moore neighborhood will contain  $(3*d - 1)$  cells where d is the number of dimensions.

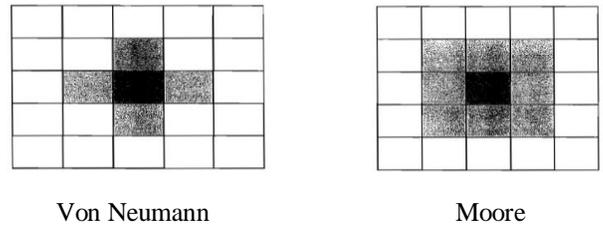


Fig. 3. Cellular automata neighborhood types

Instead of transition functions on the neighbor's states, the game of life puts two simple rules invented by the English mathematician Jon Horton Conway [16] .

These rules are:

- A dead cell becomes alive at the next generation if exactly three of its neighbors are alive.
- Alive cell at the next generation remain a live if either two or three of its neighbors are alive but otherwise it dies.

In dealing with data mining[6] , cellular automata grid represents the instance space and hence the cell state will represent the instance class. The cells are organized and connected according to attribute value ranges. The instance space will form a (multi-dimensional) grid over which the cellular automata operate. The grid will be initialized with training instances, and the CA run to convergence. The state of each cell of the cellular automata grid will represent the class assignment of that point in the instance space. The intention is that cells will organize themselves into regions of similar class assignment.

The transition function (update rule) becomes a simple voting rule that locally reduces entropy. The voting rule examines a cell's neighbors and sets the cell according the number of neighbors that are set to a given class. In this work we use the von Neumann neighborhood because it is linear in the number of dimensions of the instance space, so it scales well.

This is implemented in a transition rule called n4V1[2] . This rule examines each cell's four neighbors and sets its class to the majority class. It can be a stable rule, in that, once the cell's class has been set it will not be changed. It also can be nonstable, where the cell state can change along the iterations if the majority class of the neighborhood changed.

The global effect of the n4V1 update rule is that each cell in the grid becomes assigned with the class of the nearest initial point as measured by Manhattan distance. The Manhattan distance aspect stems from the fact that the CA uses the von Neumann neighborhood, so each cell's influence spreads outward along the two dimensions. The first neighbor of a cell that changes state, from empty to a class, will result in that its neighborhood cells changing state in the next time step[6] . If 0 means that the cell is empty and the instance's classes are 1 and 2 then the n4V1 nonsatble update rule will look like:

$$n4V1 \text{ nonstable} = \begin{cases} 0 : \text{class1 neighbors} + \text{class2 neighbors} = 0 \\ 1 : \text{class1 neighbors} > \text{class2 neighbors} \\ 2 : \text{class1 neighbors} < \text{class2 neighbors} \\ \text{rand}(\{1,2\}) : \text{class1 neighbors} = \text{class2 neighbors} \end{cases} \quad (7)$$

where rand (1, 2) selects randomly from the elements with equal probability.

This was the Boolean cellular automata where the cell either alive or dead (has a class or not). But in the case of uncertainty, the fuzzy cellular[12] is more efficient as its cells contain the state (class) and the membership degree of that class. This will help handle uncertain data where the variable are fuzzy and hence producing fuzzy classes.

Fuzzy cellular automata (FCA) [12] are continuous cellular automata where the local rule is defined as the “fuzzification” of the local rule of a corresponding Boolean cellular automaton in disjunctive normal form[11]. The “fuzzification” is accomplished by using the fuzzy extension of the Boolean operators AND, OR and NOT. Depending on which fuzzy operator is used, a different type of Fuzzy cellular automata can be defined. Among the various possible choices for the fuzzy operators, we consider the following:  $(a \vee b)$  is replaced by  $(\max(a, b))$ ;  $(a \wedge b)$  by  $(\min(a, b))$ , and  $(\neg a)$  by  $(1 - a)$ . The resulting local rule becomes the fuzzy real function that generalizes the original function. Then the n4V1 rule can be fuzzified by applying the fuzzy operators.

#### IV. THE PROPOSED FRAMEWORK

The framework proposed by this paper is to generate a fuzzy rough rule based system and enhance it using the fuzzy cellular automata. The fuzzy rough rules are simple if-then rules but with fuzzy rough variables. This research tries to build these fuzzy rough rules in three phases. The first phase is to generate the membership function for the subsets of the fuzzy variables. The second phase is to reduce the features using the fuzzy membership dependency between the features and the data set. The third phase is to design the fuzzy rough rule by summarizing the data of the reduced features basing on the rough set theory then tests these rules efficiency by the test data set. The proposed framework that outlines the main modules is represented in figure 4. The data used in the training process as well as the features data collected from experts are used as inputs to the Generating fuzzy membership function for features subsets process which outputs a data file that contains the values of the fuzzy variables and their corresponding membership degrees in the subsets of these variables. These fuzzy membership degrees and the training data again are used as inputs to the reducing features process that measures the attributes dependency and produces the core attributes. The generating fuzzy rough rules process takes the core attributes (reduced training data set) and summarizes the data set to output the corresponding fuzzy rough rules set after testing them by the test data records. The set of the fuzzy rough rules are used as an initial state for the fuzzy cellular automata parallel system to enhance the rule based system and produce what can be said as an equation of the system on time sequence. The main components are:

**Generating fuzzy membership function for features subsets:** This module uses the SOFM capabilities of unsupervised learning and clustering to generate the membership functions of the features subsets.

**Reducing Features:** This module uses the fuzzy rough attribute reduction (FRAR) algorithm to reduce the features basing on the measuring the dependency membership degree between the fuzzy variables and the training data set to produce the reduct (core attributes).

**Generating fuzzy rough rules:** This module summarizes the data set using the reduced data set according to the fuzzy rough theory to generate the corresponding fuzzy rough rules. These rules are applied on the testing data set to measure its accuracy rate.

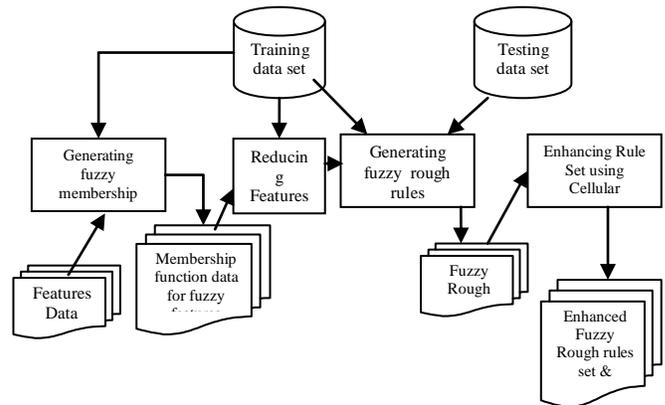


Fig. 4. the proposed framework for uncertain Knowledge

**Enhancing Rule Set using Fuzzy Cellular Automata:** this module takes the fuzzy rough rule set as an initial state and iterate according to the fuzzy n4V1 nonsatble update rule to produce the view of the fuzzy rough rule based system in the time sequence. These views of the system may give good recommendations for the experts in the field of the data under consideration (training data) and cover all the data space with accurate rule for more classification or prediction issues.

##### A. Preparing Fuzzy Features (variables)

The process of generating membership functions used to be in two phases. The first phase generates the proper clusters of the feature data. Then, the fuzzy membership function is generated according these clusters. But it is possible to generate the membership functions in one phase by combining the variable labels with the variable values in the input layer of the SOFM[4]. As mentioned the SOFM learns from its input vector so the input vector will be  $X_n = (v, S_1, S_2, \dots, S_d)$  where  $v$  is the value of the feature under consideration and  $S_1, S_2, \dots, S_d$  are the subsets that the feature will be divide to. These input vectors will be the training data and can be got by asking an expert some questions about the features like do you think that the value  $v$  of feature  $f$  belongs to  $s_1$  or  $s_2$  or  $\dots, s_d$ ? The SOFM then goes through the learning phase and update its weights according to the learning procedure mentioned in section

III.A. After convergence the weights of the SOFM will be the values of the variable and its corresponding membership degrees of the labels (subsets) in the input layer. This technique is illustrated before in a research for Generating fuzzy membership function with self-organizing feature map by Chih-Chung Yang, N.K. Bose.

Example 1 [4] is a graphical example for illustrating the technique. Suppose that the fuzzy subsets for a variable like height would be ‘short’ and ‘tall’ and the membership function for each subset is to be found. The dataset could be collected by asking a question “Do you think a person with height 6 feet is tall or short?” After dataset is collected, the labeling information may be represented by 2-D unit vectors  $[1\ 0]^T$  and  $[0\ 1]^T$  for fuzzy variable ‘short’ and ‘tall’, respectively. In the training phase of SOFM, the input feature height was combined with the labeling information to form a 3-D vector, which would be the input training sample for the SOFM.

Suppose there are five neurons in the output layer as in figure 5 and the associated weights after training process are listed in Table 1.

The fuzzy membership functions for the fuzzy variables tall and short are illustrated in figure 6. From this example the feature value height=5 has a degree of membership 0 in the subset tall and a degree = 1 in the subset short.

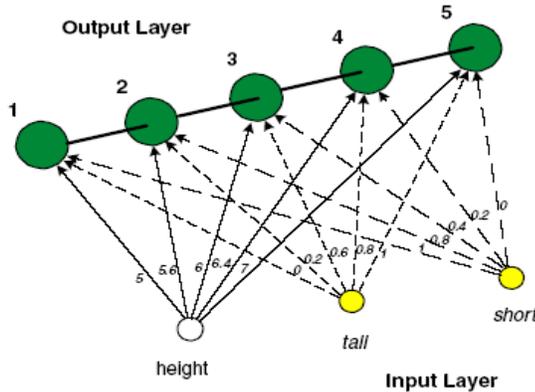


Fig. 5. [4] SOFM after training process

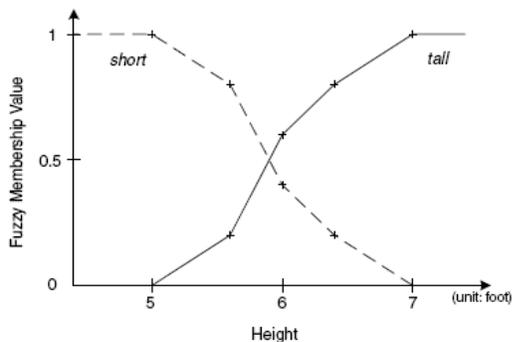


Fig. 6. [4] : Fuzzy membership function for fuzzy subsets tall and short

Neuron index	Associated weights	Feature height	Fuzzy variable	
			Tall	Short
1	$[5\ 0\ 1]^T$	5	0	1
2	$[5.6\ 0.2\ 0.7]^T$	5.6	0.2	0.7
3	$[6\ 0.6\ 0.4]^T$	6	0.6	0.4
4	$[6.4\ 0.8\ 0.2]^T$	6.4	0.8	0.2
5	$[7\ 1\ 0]^T$	7	1	0

B. Reducing Features

The reduction process aims to find the set of core attributes which the training data equivalent classes depend fully on them. Starting with the core attributes as an empty set, it can be found by calculating the dependency between all the attributes and the equivalent classes one at a time and add the attribute that maximizes the dependency degree to the core attribute set. This process is repeated until the dependency come to its highest level (usually 1). The problem, for the fuzzy attributes, is that the attributes must go through a discretization process to calculate the dependency and this causes information loss. The fuzzy Rough Set Reduction[30] solved this problem by calculating the fuzzy membership dependency degree between the fuzzy variables (attributes) and the fuzzy equivalent classes. For D is the set of equivalent classes on the universe U and A is an attribute in the fuzzy rough attributes set, the dependency can be calculated using the following equation:

$$\gamma'_{A}(D) = \frac{|\mu_{pos_A(D)}(x)|}{|U|} = \frac{\sum_{x \in U} \mu_{pos_A(D)}(x)}{|U|} \quad (8)$$

This dependency equation takes the membership degree of an object x to the fuzzy positive region of the fuzzy rough attribute A as a parameter which is:

$$\mu_{pos_A(D)}(x) = \sup_{x \in U/D} \min(\mu_{F_i}(x), \mu_{pos_A}(F_i)) \quad (9)$$

where

$F_i$  is the fuzzy attribute subsets in D (equivalent classes);

$\mu_{F_i}(x)$  is the membership degree of object x in  $F_i$ ;

$\mu_{pos_A}(F_i)$  is the fuzzy positive region of a fuzzy equivalence class  $F_i \in U/A$  which can be defined as:

$$\mu_{pos_A}(F_i) = \sup_{x \in U/D} (\mu_x(F_i)) \quad (10)$$

where

$$\mu_x(F_i) = \inf_{x \in U} \max\{1 - \mu_{F_i}(x), \mu(x)\} \quad (11)$$

where  $\mu(x)$  are the membership degrees for the fuzzy attribute subsets (fuzzy membership function). The fuzzy attribute can be declared by its fuzzy membership function. Figure 2 shows a fuzzy attribute with three subsets short, medium and tall.

This process is repeated for each attribute and the one with the biggest dependency is added to the core set until no further increasing in the dependency is accomplished. In the case of uncertain knowledge, this process can never reach the absolute dependency so a threshold can be taken to define the level of certainty accepted taking in account the time and space complexity of the algorithm. The algorithm along with an illustrating numerical example can be found in[30].

C. Generating Fuzzy Rough Rules

In the Rule based system, the Rules are the system equation so finding the best accurate rules is the only way to make an accurate system. The fuzzy rough set theory is used to get the reduct fuzzy attributes and remove the unnecessary ones but it still a data set and we need to find the necessary rules to represent it. Data summarizing is an efficient way to get the if then rules but how could fuzzy attributes be summarized hence each value is represented by the corresponding membership degrees. This process of summarizing will be only used to get the if-then rules but the membership degrees will be reserved by the attribute membership function and the data set attribute values. The next tables illustrate the whole summarizing process step by step from the data (iris data set) before reduction to the if then rules.

Table 4:Data Set Attribute Fuzzy Membership Degrees

Petal length	Petal width	class
[1; 0; 0]	[0.92;0.07;0.0003]	1
[1; 0; 0]	[0.07;0.92;0.0003]	1
[0;0.99;0.01]	[0.0033;0.8212;0.175]	2
[0.00024;0.4592;0.5405]	[0.0033;0.8212;0.175]	2
[0.0008;0.603;0.3952]	[0.003;0.070;0.929]	3
[0.00024;0.4592;0.5405]	[0.003;0.070;0.929]	3

These membership degrees can be summarized as previously illustrated in the form of 0's and 1's as in table 5.

Table 5:Data Set Attribute Fuzzy Subset summarize

petal length	petal width	Class
[1; 0; 0]	[1; 0; 0]	1
[1; 0; 0]	[0; 1; 0]	1
[0; 1; 0]	[0; 1; 0]	2
[0;0;1]	[0; 1; 0]	2
[0; 1; 0]	[0;0;1]	3
[0;0;1]	[0;0;1]	3

These tables can then be transformed into fuzzy rough if then rules for the entire data set as follows:

Table 2:Data Set Attribute Real Values

sepal length	sepal width	petal length	petal width	class
5.7	4.4	1.5	0.4	1
5.4	3.9	1.3	0.4	1
...				
6.4	3.2	4.5	1.5	2
6.9	3.1	4.9	1.5	2
...				
6.2	2.8	4.8	1.8	3
6.1	3.0	4.9	1.8	3

After the feature reduction process the data set should look like:

Table 3:Data Set Attribute Real Values

petal length	petal width	Class
1.5	0.4	1
1.3	1.6	1
...		
4.5	1.5	2
4.9	1.5	2
...		
4.8	1.8	3
4.9	1.8	3

The attribute real value can be represented by the tuples [M1; M2; M3] where M1, M2 and M3 are the membership degrees in the fuzzy attributes subsets coming from the fuzzy attributes membership function defined for each attribute in the preprocessing phase.

Table 6:Fuzzy rough if –then rules summarized from the data set after attribute reduction

	support	confidence
petal length is short & petal width is short then Iris-setosa	0.45	0.99
petal length is short & petal width is medium then Iris-setosa	0.013	0.997
petal length is medium & petal width is medium then Iris-versicolor	0.213	0.998
petal length is tall & petal width is medium then Iris-versicolor	0.12	0.529
petal length is medium & petal width is tall then Iris-virginica	0.106	0.47
petal length is tall & petal width is tall & then Iris-virginica	0.093	1

D. Enhancing System Equation by Fuzzy cellular automata

The set of fuzzy rough rules are the system equation and can be represented graphically on a two or three dimension grid. The grid dimensions are decided by the number of conditional attributes in the core set. The grid coordinates are the core attributes and cells will represent the result (class with membership rule degree inferred) of the if-then rules. The cell state will be the c & μ where the c is the class index and μ is the membership degree of that class (calculated initially from the fuzzy rules using the simple inference methods). During the iterations of the fuzzy cellular automata parallel system, the cell state will be formed according to the fuzzy n4V1 nonstable update rule (transition function). This transition function is a fuzzification of the regular n4V1

nonstable update rule which resulted by replacing the Boolean operators AND, OR and NOT by their fuzzy extensions. The update rule decides the cell state at the next time step  $[C_i(t+1) \& \mu_{ci}(t+1)]$  basing on the cell and the cell's neighbors states at the current time step.

$$\text{Fuzzyn4 V1 nonstable} = \begin{cases} [0 \& 0] : \sum_i \mu_{ci}(t) = 0 \\ [\max_{\mu_{ci}}(ci) \& \max_{\mu_{ci}}(\mu_{ci}(t)) : \mu_{ci}(t) \text{are\_not\_equal} \\ [\text{rand}_i(i) \& \text{rand}_{\mu_{ci}}(\mu_{ci}(t)) : \mu_{ci}(t) \text{are\_equal} \end{cases} \quad (12)$$

Where  $\mu_{ci}(t)$  is the class membership degree of the class  $i$  at time  $t$ .

The membership degree at time step  $t$  is the conjunction of all the neighbors' membership of the same class index:

$$\mu_{ci}(t) = \min(\mu \text{ class\_i\_neighbors}(t)) \quad (13)$$

or we can take the average instead of the min functions:

$$\mu_{ci}(t) = \text{avg}(\mu \text{ class\_i\_neighbors}(t)) \quad (14)$$

## V. EXPERIMENTAL RESULTS

The proposed hybrid model is composed of four main sub modules that are implemented using soft computing techniques. The first sub model is the Generating fuzzy membership functions for features subsets which is responsible for generating the degree of membership of the values of the variables in their corresponding subsets. This process uses the SOFM, which uses its unsupervised learning and clustering ability to learn the weights of the neural net from the input training data vectors, to obtain the variables values and their corresponding membership degrees. Using these values we can draw an analogue membership function for each subset of the variables. This technique is implemented before in a previous work [4] and we just make use of it to prepare the fuzzy variables for the generating fuzzy rough rules process. The second sub module is the attribute reduction process which makes use of the fuzzy rough dimensionality reduction process to measure the membership dependency degree between the whole data set and each attribute. This process picks the set of attributes that maximizes the dependency with the minimum loss of information. This set of attributes are said to be the core attributes which is used later in the third sub module to generate the set of fuzzy rough rules which represent the system. The third process is the generating fuzzy rough rules module that applies the summarizing technique to transform the data set under the core attributes into a set of fuzzy rough if-then rules. The fourth module is the rule enhancement which makes use of the parallel distributed processing of the fuzzy cellular automata system to generate the system equation on time series. The fuzzy cellular takes the set of fuzzy rough rules generated in the third process as its initial generation and use the fuzzy update rule (fuzzy transition function) to generate new fuzzy rough rules

its corresponding membership function. These new fuzzy rough rules could be used by experts to examine the system behavior in time steps and to make the system grid visible to see which class is most probable in the direction of which attribute. This could be thought of as a prediction system as well as a classification system.

In experiments, SOFM is used to generate the membership function of each of the attributes in the data set. The SOFM is trained with 3 or 4 input neurons (one for the feature value and the rest for the subsets) depending on the number of subsets of the features and 15 output neurons that produced 225 feature values and their corresponding membership degrees. These values were used to draw an analogue function for each feature. Figure 7 shows the membership functions of the 9 conditional features (before reduction) of the breast cancer data set. These membership functions along with the data set goes through the fuzzy rough dimensionality reduction process that measures the attributes dependency and selects the core attributes that represent the data set and reduce the data set accordingly. This process depends on some threshold which defines the acceptable loss of information. In some data sets this threshold could be 0 and in others could be 0.01 or 0.001. The threshold is selected after applying more than one trial of the reduction process and selecting threshold that reduces the error rate.

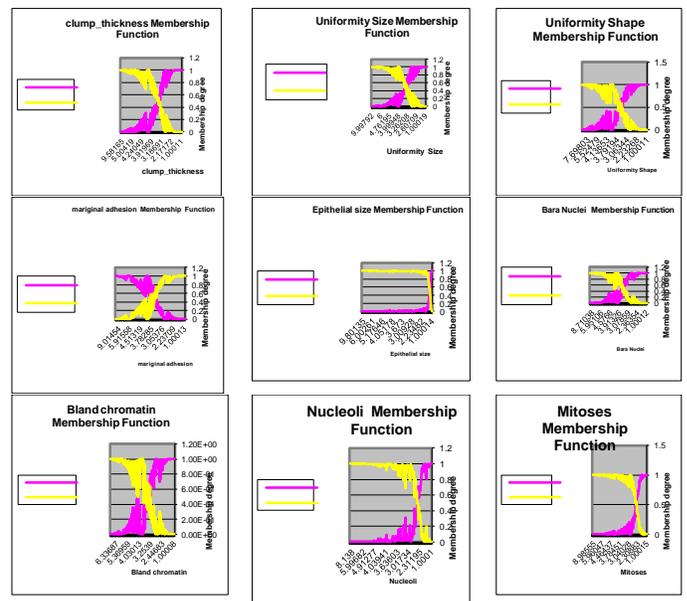


Fig. 7. Membership functions of the liver data features from the SOM process

The reduced data set (data of the core attributes only) is summarized according to the technique illustrated in section IV.C. The set of summarized fuzzy rough if-then rules are tested against a test data set to measure the overall accuracy of the system equation (if-then rules). After the whole rule generation process is finished, the if-then rules are used as the initial generation in the fuzzy cellular grid. The cell states are used as  $c_i \& \mu$  where the  $c_i$  is the rule output ( 2 or 3 classes in experiments) and  $\mu$  is the rule membership degree that

corresponds to the cell. The fuzzy transition function (fuzzy update rule) illustrated in section IV.D is used to generate fuzzy rough rule based system in time steps basing on the values of the cell's neighbors states. After convergence, the cellular grid plots the system equation which represents the system behavior.

The breast cancer fuzzy rough rule based system has a core of two attributes (the reduct is Uniformity Size and Epithelial Size). The system two dimensions grid resulting from the fuzzy cellular enhancement process along with the initial generation are illustrated in figure 9. The blue is the benign class and the color degrees are to which degree the class is accurate (membership degree of the class according to the rule membership inference) and the red is the malignant class and also the color degrees are the membership degrees. The grids are plotted by mat lab and the color bar in the figures represents the coloring memberships. The data sets used in this research to test the model are taken from the UCI machine learning repository and their properties are illustrated in table 7. The data set records are divided in two equal parts (one for the training data and one for the test data).

Name of the data set	No of attributes	No of continuous attributes	No of categorical attributes	No of data records	No of classes
weather	4	2	2	14	2
Breast Cancer	10	10	0	699	2
Wine	13	13	0	168	3
liver	6	6	0	345	2
Iris	4	4	0	150	3

The comparison between the proposed model and other techniques is listed in table 8 which shows the accuracy levels of the rule sets generated by C4.5, Neural Networks, Naïve Bays, SOFM&PGA[22] and the proposed model (SOFM + Fuzzy Rough) on five different data sets. Figure 8 shows the same comparison in graphical mode.

These comparisons show that the proposed hybrid model gave better accuracy level than the previous ones. The results indicate an average accuracy of around 87%, with accuracies above 60.1% even for quite small training sets. This compares favorably with previous systems for classifying the same data sets, whose average accuracy is 80%.

	c4.5	neural	naïve Bays	SOFM +PGA	SOFM + Fuzzy Rough
iris	84.5	91.2	88	81.3	89.3
Weather	68	78	64	71.4	100
liver	49	47	51	60.7	60.1
Breast Cancer	95.1	95.7	95.7	94.2	97
Wine	82	89	88	64.3	88.5

iris	84.5	91.2	88	81.3	89.3
Weather	68	78	64	71.4	100
liver	49	47	51	60.7	60.1
Breast Cancer	95.1	95.7	95.7	94.2	97
Wine	82	89	88	64.3	88.5

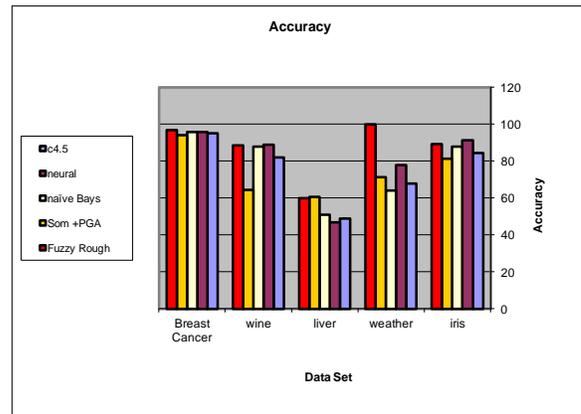
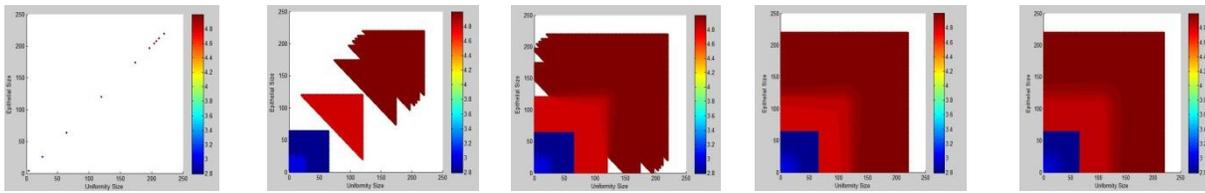


Fig. 8. The accuracy of the rule set of the proposed model and some other rule generator algorithms

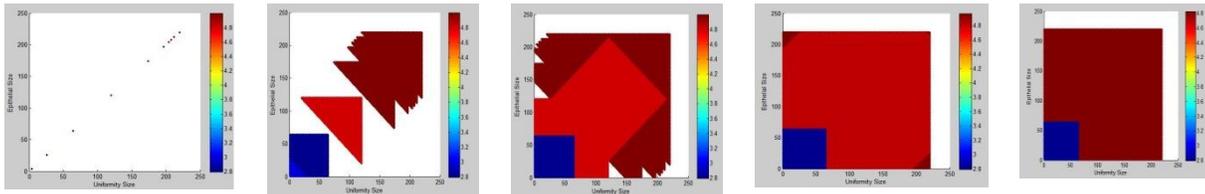
## VI. CONCLUSION

Knowledge based systems are very important tools in data mining field as they extract knowledge from concrete uncertain data. Rule based systems are a branch from knowledge based systems that depend on a set of rules to extract the information (consequences) from data (propositions). The uncertainty in the data requires a special kind of rules. Fuzzy rough rules are concerned with uncertain propositions and consequences (variables). These uncertain propositions and consequences can be represented by fuzzy variables which deal with uncertainty by means of membership functions declared for the variable values. Defining good fuzzy variables is important for generating efficient fuzzy rough rules.

Designing fuzzy rough rule based system that deals with uncertainty contains a number of problems such as declaring the variables membership functions, reducing the features (variables) to get the core attributes and generating the set of fuzzy rough rules. These problems can be solved by soft computing techniques so as to automate the whole knowledge based system. The system can help in extracting knowledge but only according to the space that the rules cover. Enhancing the fuzzy rule based system is another problem that needs to be automated and the system will be suitable to cover the whole data space and give future recommendations for experts as well.



Initial grid and the fuzzy cellular output grids after 100,200,300 and 400 iterations using the equation No. 14



Initial grid and the fuzzy cellular output grids after 100,200,300 and 400 iterations using the equation No. 13

Fig. 9. The breast cancer fuzzy rough rule grids in time series.

This paper is concerned with solving the problems which face the knowledge based framework automation process by soft computing algorithms. The first problem is declaring fuzzy variables and their corresponding membership functions. SOFM are used in clustering its inputs. This property can be used efficiently in clustering the variables into the correct subsets and produce the representative membership functions. The second problem is the dimensionality problem (reducing the feature set size). The fuzzy rough attribute reduction (FRAR) algorithm measures the dependency membership degree between the fuzzy variables and the data set basing on the fuzzy-rough set theory and this will help in generating the core attributes (reduct) and solving the dimensionality reduction problem. The third problem is the fuzzy rough rule generation which is solved by the data summarizing process which find the rules from removing redundancy and repetition from the data sets. Solving these problems contributes in producing an efficient and accurate fuzzy rough rule based system which represents the knowledge based framework which this research intended to make. This system is accurate according to the accuracy measure process by calculating the error rate of the system when applied to a test data set. In an enhancement process to the fuzzy rough rule based system, fuzzy cellular automata parallel system is used to generate the system states in time series. Using the system's set of fuzzy rough rules (which can be thought of as the system equation) as the initial state and the fuzzy update rule (fuzzy transition function) which is a fuzzy equivalent to the n4v1 non stable update rule, the fuzzy cellular automata iterate to produce the system equation on time series, cover the whole data space with accurate rules and also can give suitable recommendations about the data set under consideration to the experts.

The experimental results illustrated the framework development process and the parameters used in each step. The comparisons between the proposed framework and other systems on different data sets proved that the proposed framework is accurate and stable even for small size data sets.

The fuzzy rough rule based system grid for the breast cancer data set is presented on its initial state and in time series after the cellular automata parallel system worked on it.

#### REFERENCES

- [1] A. A. Freitas, "Data Mining and Knowledge Discovery with Evolutionary Algorithms", Springer,Berlin, 2002.
- [2] A. Piwonska and F. Sereydnski, "Solving Two-Dimensional Binary Classification Problem with Use of Cellular Automata", in *AUTOMATA the 17<sup>th</sup> International Workshop on Cellular Automata and Discrete Complex Systems Proceedings*, Santiago, Chile, 2011.
- [3] B. PLACZEK, "Fuzzy cellular model for traffic data fusion, Transport problems", volume 4 issue 4, pp. 25-35, 2009.
- [4] Chih-Chung Yang, N.K. Bose, "Generating fuzzy membership function with self-organizing feature map", *Letters Volume*, 1, Pages 356–365, April 2006.
- [5] D. E. Goldberg, "Genetic algorithms in search, optimization, and machine learning", Addison-Wesley, 412, 1989.
- [6] Fawcett, "Data mining with cellular automata", *ACM SIGKDD Explorations Newsletter*, 10(1): pp. 32–39, 2008.
- [7] Fevrier Valdez, Patricia Melin and Herman Parra, "Parallel genetic algorithms for optimization of Modular Neural Networks in pattern recognition", *IJCNN* ,pp.314-319,2011.
- [8] Fevrier Valdez, Patricia Melin and Oscar Castillo, "Evolutionary method combining Particle Swarm Optimisation and Genetic Algorithms using fuzzy logic for parameter adaptation and aggregation: the case neural network optimization for face recognition", *IJAISC*, Vol.2(1/2), pp.77-102, 2010.
- [9] Fevrier Valdez, Patricia Melin and Oscar Castillo, "An improved evolutionary method with fuzzy logic for combining Particle Swarm Optimization and Genetic Algorithms". *Appl. Soft Comput.*, Vol.11(2), pp.2625-2632, 2011.
- [10] Fevrier Valdez, Patricia Melin and Oscar Castillo, "Evolutionary method combining Particle Swarm Optimisation and Genetic Algorithms using fuzzy logic for parameter adaptation and aggregation: the case neural network optimization for face recognition", *IJAISC*, Vol.2(1/2), pp.77-102, 2010.
- [11] H. Betel and P. Flocchini, "On the Relationship between Boolean and Fuzzy Cellular Automata", 2009.
- [12] H. Ishibuchi, K. Nozaki and H. Tanaka, "Adaptive Fuzzy Rule-Based Classification Systems", *IEEE Trans. on Fuzzy Systems*, vol. 4, no. 3, pp. 238-250, 1996.
- [13] H. Ishibuchi, T. Nakashima and T. Murata, "Performance Evaluation of Fuzzy Classifier Systems for Multi-Dimensional Pattern Classification

- Problems", IEEE Trans. Syst., Man, Cybern, Part B, vol. 29, pp. 601-618, 1999.
- [14] H. Ishibuchi, T. Nakashima and T. Murata, "A Fuzzy Classifier System that Generates Fuzzy If-Then Rules for Pattern Classification Problems", Proc. of 2nd IEEE Int. Conf. Evolutionary Computation, Perth, Australia, pp. 759-764, Nov. 29-Dec. 1, 1995.
- [15] Janusz Kacprzyk, "Studies in Fuzziness and Soft Computing" ,ISBN 978-3540737223, ISSN: 1434-9922 (Print) 1860-0808 (Online), Springer Berlin / Heidelberg ,2009.
- [16] J.L. Schiff, "*Cellular Automata A Discrete view of the world*. 2008: Jhon Wiley & SONS INC. publications
- [17] J. Virant and N. Zimic , "Fuzzy automata with fuzzy relief", IEEE Trans. Fuzzy Systems, Vol. 3, No. 1, pp. 69-74, 1995.
- [18] Kenji Suzuki, "Artificial Neural Networks: Methodological Advances and Biomedical Applications", InTech, ISBN-13: 9789533072432, 2011.
- [19] Lotfi A. Zadeh, "From computing with numbers to computing with words —from manipulation of measurements to manipulation of perceptions", in International Journal of Applied Math and Computer Science, pp. 307-324, vol. 12, no. 3, 2002.
- [20] Lotfi A. Zadeh, "Fuzzy sets and systems". In: Fox J, editor. System Theory. Brooklyn, NY: Polytechnic Press, pp. 29-39, 1965.
- [21] M. Vose., "The Simple Genetic Algorithm Foundation and Theory", MIT Press, 251, 1999.
- [22] Mona Gamal, Ahmed Abo El-Fatoh, Shereef Barakat and Elsayed Radwan, "A Hybrid of Self Organized Feature Maps and Parallel Genetic Algorithms for Uncertain Knowledge", International Journal of Computer Applications (0975 – 8887) Volume 60– No.6, December 2012.
- [23] Nan-Chen Hsieh , "Rule Extraction with Rough-Fuzzy Hybridization Method", Advances in Knowledge Discovery and Data Mining; Lecture Notes in Computer Science, Vol. 5012, pp 890-895, 2008.
- [24] O. Cordon, F. A. C. Gomide, F. Herrera, F. Hoffmann and L. Magdalena, "Ten Years of Genetic Fuzzy Systems: Current Framework and New Trends", Fuzzy Sets and Systems, Pages 5-31, 2004.
- [25] Oded Maimon and Lior Rokach , "Soft Computing for Knowledge Discovery and Data Mining" , ISBN-10: 0387699341, ISBN-13: 978-0387699349, Springer; 2008 edition, November 26, 2007.
- [26] O. Cordon, F. Gomide, F. Herrera, F. Hoffmann, and L. Magdalena, "Ten years of genetic fuzzy systems: Current framework and new trends", Fuzzy Sets and Systems, pp. 5-31, 2004.
- [27] Saroj, Nishant Prabhat," A Genetic-Fuzzy Algorithm to Discover Fuzzy Classification Rules for Mixed Attributes Datasets", International Journal of Computer Applications, Vol 34– No.5, November 2011.
- [28] S. P. Tiwari and Arun K. Srivastava, "Fuzzy rough sets, fuzzy preorders and fuzzy topologies Fuzzy Sets and Systems", Vol. 210 , pp. 63-68, January 2013.
- [29] T. Kohonen, "Self-Organizing Maps", Springer Series in Information Sciences, Vol. 30, Springer, Berlin, Heidelberg, New York, ISBN 3-540-67921-9, ISSN 0720-678X 1995, 1997, 2001.
- [30] R. Jensen and Qiang Shen, "Fuzzy-Rough Sets for Descriptive Dimensionality Reduction", Proceedings of the IEEE International Conference on Fuzzy Systems, Vol: 1 ,Pp 29 – 34, 2002.
- [31] Y. Caballero, D. Alvarez, R. Bello and M. M. Garcia, "Feature Selection Algorithms Using Rough Set Theory, In Intelligent Systems Design and Applications", ISDA Seventh International Conference on, pp. 407-411 , 2007.
- [32] Z. Michalewicz, "Genetic Algorithms + Data Structures = Evolution Programs", Springer-Verlang, 252, 1992.