A Big Bang – Big Crunch Type-2 Fuzzy Logic Based System for Fraud-Detection: Case Study Balad Bank in Sudan

Saeed Khalil Saeed [1] and Hani Hagras [2]

[1] Department Computer Science, Sudan University of Science and Technology (SUST), Sudan
[2] The Computational Intelligence Centre, University of Essex, United Kingdom
saeed_kl@hotmail.com

Received:07/04/2020
Accepted: 13/06/2020

ABSTRACT– Improved fraud detection systems are vital tools for the evolution of the Sudanese banking sector where the traditional fraud detection models are incapable of overcoming the emerging, innovative and new attacks that threaten large financial institutions. Hence, there is a need for accurate and transparent techniques which can automatically detect fraud with high speed and identify its causes and common patterns. Many of the Artificial Intelligence (AI) techniques are effective and provide good predictive models. Nevertheless, they are considered as black-box models. On the other hand, the white box models are easy to understand and analyze, but result in a large number of rules, besides having many parameters in each rule. In this paper, we present a novel system based on the Big Bang–Big Crunch optimization (BB–BC) approach, which is combined with type-2 Fuzzy Logic Systems to result in a small set of short IF-Then rules for the fraud detection within the Sudanese banking sector. The proposed system uses real-world dataset from Balad Bank – Sudan, which contains 803,386 transactions with 107 fraud transactions. Hence, the positive class (frauds) rate is 0.0133% of all transactions. The experimental results demonstrate that the performance of proposed system is effective in tuning the parameters of the rule base and membership functions of the Type-2 FLSs (T2FLSs) to improve the accuracy, where the proposed T2FLSs outperformed the Type-1 FLSs (T1FLSs) counterpart, as well as each rule can be simply explainable. Therefore, this can be very helpful for the Sudanese banks to start tracking the fraud cases.

Keywords: Big Bang–Big Crunch, Type-2 fuzzy logic system, fraud detection, online payments, credit cards, debit cards.

الستخلص - أنظمة كشف الاحتيال المحسنة هي أدوات مهمة لتطور القطاع المصرفي السوداني حيث النماذج التقليدية للكشف عن الاحتيال غير قادرة على التغلب على الهجمات الناشئة والمبتكرة والجديدة التي تهدد المؤسسات المالية الكبيرة. وبالتالي ، هناك حاجة إلى تقنيات دقيقة وشفافة يمكنها اكتشاف الاحتيال تلقائيًا بسرعة عالية وتحديد أسبابه وأنماطه الشائعة. الكثير من تقنيات الذكاء الاصطناعي تعتبر فعالة و تمثل نماذج تنبؤية جيدة. ومع ذلك ، تعتبر نماذج الصندوق الأسود. من ناحية أخرى ، من السهل فهم وتحليل نماذج الصندوق الأبيض ولكنها تنتج عدد كبير من القواعد ، بالإضافة إلى انها تضم العديد من الخصائص في كل قاعدة. في هذه الورقة ، نقدم نظامًا جديدًا يعتمد على نهج Big Bang - Big Crunch Optimization (BB – BC)، والذي يتم دمجه مع النوع 2 Fuzzy Logic Systems للحصول على مجموعة صغيرة من قواعد IF-Then مختصرة وذلك للكشف عن الاحتيال داخل القطاع المصرفي السوداني. يستخدم النظام المقترح مجموعة بيانات واقعية من مصرف البلد - السودان ، والتي تحتوي على 803.386 معاملة مع 107 معاملات احتيال. وبالتالي ، فإن المعدل الإيجابي (الاحتيال) هو 0.0133٪ في جميع المعاملات. توضح النتائج التجريبية أن أداء النظام المقترح فعال في ضبط معاملات قاعدة الاحكام ووظائفها العضوية من النوع 2 FLS (T2FLSs) وذلك لتحسين الدقة ، حيث تفوقت T2FLSs المقترحة على النوع الاول من نظيره FLS (T1FLSs)، وكذلك كل قاعدة يمكن تفسيرها ببساطة. لذلك يمكن أن يكون هذا مفيدًا للغاية للمصارف السودانية لبدء تتبع حالات الاحتيال.

INTRODUCTION

Many forms of fraud threaten the financial institutions, whereas the concept of fraud in financial systems includes several types of illegal activities, or unauthorized transactions, such as online banking fraud (Remote Banking), falsification of documents, phishing, fraudulent loans, fraudulent accounts, cheque, debit/credit card fraud (Payment Cards) etc.

In 2018 [1] payment card fraud losses worldwide reached $27.85 billion, and were up from $23.97 billion the year before. Such fraud losses are expected to increase to $35.67 billion in the next

five years and $40.63 billion in next 10 years. In [2] Fraud losses are incurred by:

Merchants, payment card issuers, acquirers of card transactions from merchants, and acquirers of card transactions at ATMs on all credit, debit, and prepaid general purpose. One of the most common type of fraud is a debit card fraud, which occurs when an attacker obtains access to an owner's debit card number and, in some cases, personal identification number (PIN), to make unauthorized purchases or withdraw cash from a client's account via ATMs or POS.

In Sudanese banks, there is an e-payment claim section or department, which is responsible for all kind of disputes or claims as well as the ATM disputes, in other words the claims issue by the customer when he finds one or more debits form his account, but he does not think he received it. Nevertheless, there are no statistics or studies about the debit card frauds in Sudan, usually banks do not prefer to reveal fraud cases due to marketing reasons.

There are many AI fraud detection algorithms which have been introduced, and most of these applications have achieved promising results, but as mentioned above, it is still very challenging to precisely and quickly detect credit card fraudulent transactions, due the difficulties of accessing a real-world dataset and to deal with data imbalance.

Our previous work used [3] type-2 Fuzzy Logic Systems (T2FLSs) and Fuzzy C-Means Clustering (FCM), which can detect fraud in debit cards, using real world dataset extracted from financial institutions in Sudan. Extended to this work, we employ "Big Bang Big Crunch (BB-BC), which was presented in [4]. It is an intelligent optimization algorithm and can improve the accuracy and speed, as well as learned quickly.

In this paper, we introduce a new white box Artificial Intelligence (AI) approach, using BB-BC as an evolutionary computation approach to reduce the number of rules, and generate very short rules by using small numbers of features, with very high accuracy and low computation time. This will be used for financial fraud detection in Sudanese banks specially in Balad bank where all the fraud cases captured manually, starting by focusing on debit card fraud, which can be later generalized to other kinds of financial fraud.

The remainder of the paper is organized as follows: the following section presents a brief overview of a black box & white box on fraud detection systems. This will be followed by brief overview of type-2 FLSs. Then an introduction on Big Bang–Big Crunch optimization method, followed by the proposed optimization fraud detection T2FLS for the Sudanese financial sector. This will be followed by experiments and results. The last section provides the conclusions and future work recommendations.

A Black box & White box Fraud Detection Techniques:

Fraud detection applications are significant in financial sectors. Therefore, fraud detection has been the hot topic of numerous surveys and review articles. Many fraud detection techniques have been successfully applied and many categorizations have been employed. Accordingly, the AI techniques could be categorized into black box & white box [4] [5] [6]. This section presents some of AI detection algorithms, based on black box & white box categorization, as shown in Figure 1.

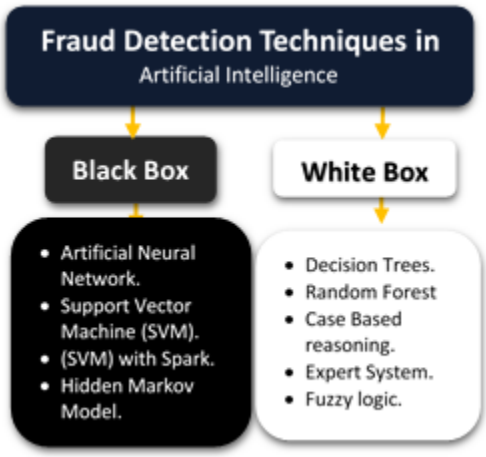

Figure 1: The categorization of fraud detection techniques

AI Fraud Detection Techniques (Black Box):

The Black-box algorithms (BB) take a series of inputs, and return equivalent outputs, while keeping internal states such as model architecture concealed [7]. Consequently, the black-box models are ambiguous. In this Section, we briefly discuss some of BB algorithms for credit card fraud detection and anomaly detection models.

Artificial Neural Network

ANN consists of three layers or more, an input layer of neurons or nodes, one or more hidden layers of neurons, and a final layer of output neurons [8].

There are two types of ANN learning methods, which are the supervised and unsupervised (using behavior method such as normal or fraudsters' behavior and no need for historical data). Or both [9], [10]. The main goal of ANN (or simply NN) is to resolve whether, or not, a transaction is fraudulent, based on hidden layers of neurons.

Generative Adversarial Networks

GAN is a generative and estimating model designed by Ian Goodfellow [11] in 2014. In a GAN system, two functions (generative G and discriminator D), represented by neural networks, are trained simultaneously. In the case where two functions are defined by hidden layers or multilayer perceptron, the whole method can be trained with backpropagation.

In [12], a generative adversarial network-based telecom fraud detection is presented at the receiving bank. This model assumes each large transaction is fraudulent, and the bank can take suitable measures to prevent potential scammers in case they exceed their threshold. To effectively train the probabilistic relationship among the input feature the algorithm uses a deep denoising auto encoder, as well as an adversarial training to accurately identify both the positive and negative models in the data.

Support Vector Machine

Support Vector Machines (SVMs) were introduced by Cortes and Vapnik in 1995 [13] to solve the classification and regression problems. In [14], they used SVM in fraud detection. The main idea of SVM is to originate an optimal hyperplane that maximizes the margin between two classes

In [15], a method to evaluate validity of new transactions is proposed to detect fraud by using Spark with SVM techniques. SVM has a good performance, but it is a slow and complex classification algorithm.

Hidden Markov Model

HMM is a statistical tool and has been used widely to detect the fraud [16] [17] [18]. At first, HMM is trained by the normal or fraudulent behavior of such, as a spending pattern or elapsed time between the current and preceding transactions. When the trained HMM receives an unaccepted transaction which has anomaly with high probability, it considered it as fraudulent transaction after making combinations of some sequences or statistical process

AI Fraud Detection Techniques (White Box):

In contrast, the White-box algorithms (WB) are considered more transparent [19]; thus, it is intelligible and easy to analyze, but it results in large numbers of rules and uses expanded rules. Some of these techniques are presented below.

Decision Tree

DT is considered as a classification tool, which is used to solve complicated problems [20][21]. In [22] they proposed a system to detect fraud in credit card transactions by using a DT with combination of Luhn's and Hunt's algorithm. In [23], a cost-sensitive decision tree method is proposed for fraud detection by using a real-world dataset.

Decision trees are easy to understand and simple to execute and capable of dealing with noisy data plus it is flexible in classification but require frequently maintenance to check the fresh leafs.

Random Forest

The random forest algorithm [24], it is a supervised learning model, for each new input the random forests combine many decision trees and aggregates their output or "vote" for a class by taking the maximum votes.

In [25], random forest is used as a classifier method for credit card fraud detection. The authors used an e-commerce company data set from China and two types of random forests were used to train the behavior features of fraudulent and genuine transactions. The first is Random-tree-based random forest used as base classifier of random forest I, and the second is CART-based random forest (Classification and Regression Trees) used as base classifier of random forest II. After the comparison of the two types of random forest, the performance of random forest II is more appropriate.

Case-Based Reasoning

CBR [26] is a model for analyzing the toughest cases and resolving problems, which have been misclassified by current techniques. CBR produces results from previous comparable cases and reuses them in different problem cases, In [27], the idea of combining algorithms to increase the power of prediction is improved. This is done by using different methods such as: diagnosis with three resolution strategies (sequential resolution, best guess and combined confidence), which analyzed the retrieved cases; and the other algorithms are:

best match algorithm, probabilistic curve algorithm, density selection algorithms, negative selection algorithms and default goal [27].

K- Nearest Neighbor algorithm (K-NN) is a clustering or classification supervised learning algorithm. It classifies any input transaction by computing majority vote between the K most similar instances of nearest point to new received transaction. Accordingly, similarity is defined according to a distance metric between two data points. So, if the nearest neighbor is fraudulent, then the transaction is marked as a fraudulent and vice versa, [28]. Figure 2 shows nearest neighbor and K- nearest neighbor, and simulates the transactions after clustering into fraud and non-fraud transactions.

Figure 2: Nearest neighbor and KNN

Expert Systems

ESs obtain knowledge from a human expert and store it in a rule-based system, such as IF-THEN rules [29], which is defined also as a representative system capable of producing a reason about some knowledge-rich domain to give advice, with a view to resolve the issues [30]. In [31], an expert system model with real data from a Canadian bank is proposed to detect the fraudulent practice of card within the authorization process.

Fuzzy logic Type-1 Fuzzy logic T1FLC is used in credit card fraud detection. In [32], a fuzzy Darwinian detection of credit card fraud is proposed as an evolutionary-fuzzy system, This technique uses two main algorithms: the first is a genetic programming (GP) and the second is a fuzzy expert system to produce fuzzy logic rules capable of classifying each incoming credit card transaction into two groups "suspicious" and "non-suspicious".

Fuzzy logic is an explainable white box technique, and has a good knowledge representation, and is maintainable due to the transparency. However, type-1 fuzzy logic cannot deal with uncertainty. The following section presents an overview of type-2 fuzzy logic system.

Brief Overview on Type-1 and Type-2 Fuzzy Logic Systems

This section discusses an introduction to T1FLSs and T2FLSs, and explains their properties. Fuzzy Logic was introduced by Lotfi A. Zadeh in 1965. Fuzzy Logic tries to mimic the way of human thinking, which is an approximate and imprecise way, such as linguistic human concepts (Kid, Young and Old), which are not precise [33][34].

Fuzzy logic may appear similar to probability and statistics as well. Moreover, both terms are used to describe uncertainty, although, fuzzy logic is different from both. Figure 3 Shows the Structure of type-1 fuzzy logic controller, which is comprised of four main parts.

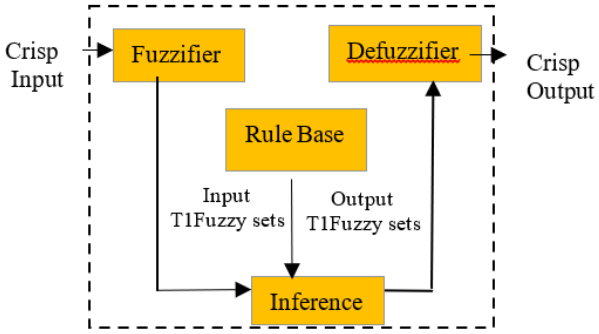

Figure 3: The Structure of T1FLC

The fuzzification role is to convert each crisp input, or measurement, into fuzzy values; there are two fuzzifier operators; a singleton fuzzifier and is used with a precise value, whereas the second operator is a probabilistic fuzzifier and can be used with imprecise measures, which means the probabilistic fuzzifier can be used when receiving input data from any inaccurate devices or sensors. It can also be used when the data is disturbed by random noise.

Rule Base or set of IF: These rules are the core of a FLC. Rules can be extracted from numerical data or can be designed by experts. These rules are fired by using inference mechanisms. Fuzzy inference machine plays a very essential role. It receives fuzzy input sets from the fuzzifier and produces fuzzy output sets to the defuzzifier. In addition, it selects the corresponding rules from the rule-base to be triggered. Defuzzification is considered as a

concluding unit in FLC. It produces crisp outputs from the fuzzy sets that appear at the output of a fuzzy inference machine.

Type-2 Fuzzy Logic Systems

Type-2 fuzzy sets are an extension to type-1 fuzzy logic system (T1FLS) for developed applications. However, the only difference is that there is no type-Reducer in a T1FLS and it employs type-1 fuzzy sets in the input and output of the FLS. Type-2 fuzzy sets are useful in circumstances where it is difficult to determine the precise membership function for a fuzzy set [35][36][37][38]. Type 1 fuzzy cannot handle the high level of linguistic and numerical uncertainties, and does not model UNCERTANITY [58].

Moreover, type-2 fuzzy logic has a Footprint of Uncertainty (FOU) located between the lower membership and the upper membership function all of these additional features are made to model and handle uncertainties. Figure 4 shows types of membership function in type 2 FLC, and an Upper Membership [38]

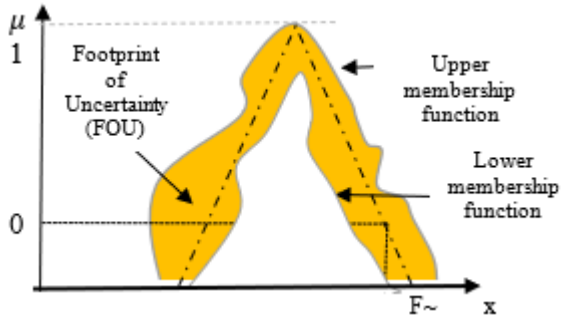

Figure 4: Membership Types of type-2 fuzzy set [38]

Many researches have proved that using interval type-2 fuzzy sets to characterize the inputs or/and outputs of FLC, has many features and advantages when compared to the type-1 fuzzy sets. The additional degrees of freedom provided by the FOU allows a type-2 FLC to produce outputs that cannot be achieved by type-1 FLCs with the identical number of membership functions [37]. There are many applications of T2FLS and has been used widely [53][54][55][56][57].

Big Bang–Big Crunch Optimization Method

The Big Bang–Big Crunch (BB-BC) algorithm is an evolutionary search approach presented by Erol and Eksin [39] [40] [41]. It relies on the evolution of the universal where it is named Big Bang – Big Crunch Theory. The BB–BC algorithm has as many benefits as other evolutionary algorithms: it does not require a clear relationship between the objective function and the constraints [42]. The steps of the BB-BC optimization can be summarized as follows [40] [41] [43]:

First Step:

In this phase, an initial generation of N candidates is generated randomly within the limits of the search space, which is called the Bing Bang phase.

Second Step:

Once the population is created, calculate the fitness values of all the candidate solutions.

Third Step:

The Big Crunch phase comes after The Big Bang phase, it comes as a convergence operator that has many inputs. Either the best fit individual or the center of mass is computed which is only one output. The center of mass can be computed as follows:

$$x_c = \frac{\sum_{i=1}^{N} \frac{1}{f_i} x_i}{\sum_{i=1}^{N} \frac{1}{f_i}} \quad (1)$$

Where the point that represents the center of mass that is denoted by x_c and the position of the candidate is x_i, whereas f_i is the fitness or cost function value of the i^{th} candidate, and N is the population size. The new candidate solutions for the next iteration Big Bang phase is normally dispersed around x_c.

Fourth Step:

After the Big Crunch phase, the algorithm must create new candidate solutions to be used as the Big Bang of the next iteration step. This by adding or subtracting a normal random number whose value decreases as the iterations elapse. This can be calculated as:

$$x^{new} = x_c + \frac{lr}{k} \quad (2)$$

Where x^{new} is the new candidate and it is upper and lower bounded, x_c the position of the center of mass, is, l is the upper limit of the parameter, r is the random number and k is the iteration step. Then if new candidate x^{new} is bigger than the upper limit l then x^{new} is set to l or if the new point $x^{new}$is slighter than the lower limit u then $x^{new}$is set to u.

Fifth step:

The algorithm continues until W number of iterations are finished or predefined stopping criteria has been met else return to the Second Step. [44][45] Presented a strategy to enhance the performance of the general BB-BC algorithm, called BB–BC truss design algorithm that follows the general procedure developed by Erol and Eksin, In this modification, the positions of new points at the beginning of each Big Bang are normally distributed around a new point located between the center of mass and the best solution:

$$x^{new} = \beta\, x_c + (1 - \beta\, x_{best}) + \frac{r \propto (x_{max} - x_{min})}{k} \qquad (3)$$

Where β is the parameter controlling the influence of the best solution x_{best} on the location of new candidate solutions. $\propto$ is a parameter limiting the size of the search space, x_{max} and x_{min} are the upper and lower limits. This adaptation outcome is depicted in a substantial enhancement in the quality of the solutions and the efficiency where the best solution influences the direction of the search. Therefore, potential progress in the computational efficiency and overall performance occurs.

Dataset Reliability and Validation:

The dataset has been acquired from Balad Bank (formerly Al-Shamal Islamic Bank) Khartoum – Sudan, after we got the authorization from the bank management therefore, we omitted any personal information in this dataset, which is considered as real-world financial dataset. Regarding the fraud cases, Bald Bank has special section responsible for the ATM card claims and disputes, this department observe and receive the suspected cases as well, it always makes deep investigation about the case to analyses the case, thus to retrieve the money, in 2016 most of fraud cases happened by loss of card, 107 fraud cases collected from this department and selected from the dataset as fraud cases.

Balad bank datasets contain transactions made by ATM cards (debit cards) in 2016 by Sudanese cardholders using banking payment channels. ATM cards also can be use in POS or in many applications that allows the customers to do many services such as bill payment, transfer money or e-commerce, accordingly all of these types of transactions included in this dataset. This dataset presents transactions that occurred in eleven months, which it contains, 107 frauds out of 803,386 transactions. The dataset is highly unbalanced; the positive class (frauds) rate is 0.0133% of all transactions.

The data has been collected from many Tables then it is stored in one Table; unfortunately, due to confidentiality issues many fields were removed and the total number of dataset features is 17. All of the features contain only numerical input so many features were converted to numbers as shown in Table 1.

TABLE 1: APPROACHES FOR CONVERTING STRING VALUES TO NUMERIC

No.	Feature	Approach for converting to numbers
1	Weekday	represents the weekdays in number from 1-7
2	Gender	1 for Male and 0 for Female.
3	Time	The time is converted to 24 hours format and the minutes to two-digit, example 6:45 pm converted to 19.45
4	Date	Represents the day in number from 1 – 31
5	Marital Status	0 for single, 1 for married, 2 for divorced and 3 for widowed
6	Class	1 for fraud and 0 for non-fraud

Some of features replaced with sequence number for instance City, bank, Occupation, service type etc. and lookup created for them because it cannot consider as fuzzy features, hence some of them could not be significant and should be ignored to accelerate the retrieving process. Figure 5 shows some features after cleansing the dataset

The dataset has been separated randomly to two datasets, the first dataset for training purpose, which represent 70% from the cleaned dataset, the second dataset for the prediction stage and represents the rest 30%. This dataset used in previous work [3] type-2 Fuzzy Logic Systems (T2FLSs) and Fuzzy C-Means Clustering (FCM).

The proposed Optimization Fraud Detection Type-2 Fuzzy Logic Based System for the Sudanese Financial Sector:

In the preceding research [3][6], we have proposed Type-2 Fuzzy Logic Based System T2FLS for fraud detection in financial applications. It starts with the training phase of extracting the rules from the database; then handling these rules by calculating the weighted scaled dominance, which is used to resolve the conflicting rules when the data is highly imbalanced.

Max. rows: 541016 | Fetched Rows: 241016 | Matching Rows:

#	INT	REF_NO	GENDER	OCCUPATION	EDUCATION	MARITAL_STATUS	CLASS	ACC_TYPE	TIME_TR	DAY_OF_MONTH	WEEKDAY	AGE	CITY	BANK	TR_TYPE	SERVIC
1		263837527	1	1	<NULL>	2	0	4	09.32	5	5	46	32	29	6	11
2		263967632	1	5	2	1	0	1	13.54	5	5	31	33	1	6	11
3		264073217	1	9	2	2	0	4	17.44	5	5	67	<NULL>	6	6	11
4		264106133	<NULL>	<NULL>	<NULL>	<NULL>	0	4	19.02	5	5	<NULL>	<NULL>	6	6	11
5		264147125	1	1	4	1	0	1	20.58	5	5	32	31	24	6	11
6		264147203	1	1	4	1	0	1	20.59	5	5	32	31	24	6	11

Figure 5: Snapshot for the dataset

because the majority class of non-fraud transaction is much greater than fraudulent transactions. T1FLS is extracted using Fuzzy C Means Clustering (FCM) and then type-2 fuzzy set with equal incremental in FOU, for each fuzzy set used. Proper results were achieved and each rule could explain why the transaction is fraudulent or not, but this white box technique can result in large numbers of rules, with many parameters in each rule. Therefore, the fraud cannot be detected instantly, or is delayed due to the vast size of rule base. In [39] [46] [47] and in this proposed system, the BB-BC is applied to select and compute the optimized rules and the MFs of our IT2FLS, for two reasons. Firstly, to increase the accuracy. Secondly, to generate a light rule base with 200 – 400 rules and with a small number of antecedents per rule, such as 3 or 4 antecedents per rule, which can accelerate the system. Consequently, each rule could be simply understood and investigated by the user.

Nevertheless, partial coverage of the whole search space occurs when working with a small rule base, besides a small number of antecedents per rule. This is because some inputs do not fire any rules from the existing light rule base. Hence the proposed system can resolve this by using the similarity measure. The overview of the proposed system is depicted in Figure 6, where in the training phase rules are extracted from training dataset, using our previous work based on interval type-2 fuzzy logic systems [3], as well as the type-2 fuzzy Membership Functions (MFs) of the inputs to the fuzzy systems, which are then learned via Fuzzy C-Means Clustering (FCM) [48], with 20% increment in FOU, which was the highest AVG recall in [3]. Finally, the parameters of the rule base of the IT2FLS and the feature parameters of the type-2 membership are optimized by the proposed method based on BB-BC algorithm.

Generation of T1 & T2 Fuzzy set from data

To generate the fuzzy set, Fuzzy c-means (FCM), clustering algorithm was used, which allows one piece of numerical data to belong to many clusters with different membership values. This algorithm has been developed by 1973, and improved by Jim [49], which is widely used in pattern recognition. It is based on minimization of the following objective function [49]:

$$J_m = \sum_{i=1}^{n}\sum_{j=1}^{c} \mu_{ij}^{m} \|x_i - v_i\|^2 , 1 \le m < \infty \quad (4)$$

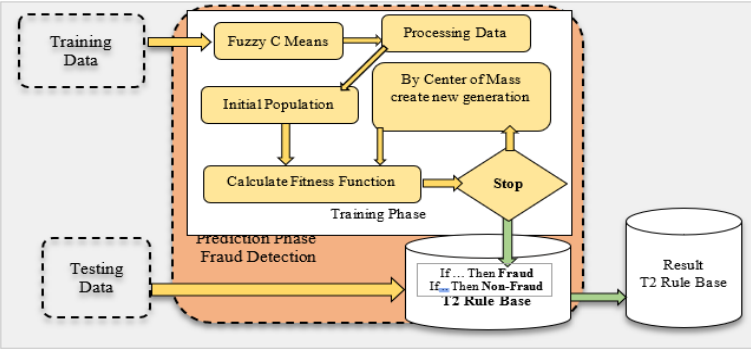

Figure 6: BB-BC Optimization for Type-2 Fuzzy Logic Fraud-Detection System for Financial Sector

where *m* is the weighting exponent and usually set to 2, μ_{ij}^{m} membership values ranging from [0,1], x_i is the *i*th of d-dimensional measured data and v_j their *d-dimension* of the cluster centers, v_j can be updated and summarized by the following equations:

$$v_i = \frac{\sum_{i=1}^{n} \mu_{ij}^{m} x_i}{\sum_{i=1}^{n} \mu_{ij}^{m}}, j = (1,2, \dots C) \quad (5)$$

Then update μ_{ij}^{m} Membership with v_i by using:

$$\mu_{ij} = \left(\sum_{k=1}^{c} \left(\frac{\|x_i - v_i\|}{\|x_i - v_k\|} \right)^{\frac{2}{m-1}} \right)^{-1}, j = (1,2, \dots C), i = (1,2, \dots N) \quad (6)$$

This iteration will stop when $\| \mu^{(k)} - \mu^{(k-1)} \| < \varepsilon$; otherwise return to equation (5), where ε is a termination criterion between 0 and 1, whereas k are the iteration steps.

A. Optimizing the rule base and Type-2 membership of the T2FLS with BB-BC:

For the purpose of employing the BB-BC, the rule base must be optimized, hence the parameters of the rule base have to be encoded into a form of a population. The type-2 rule base can be demonstrated as shown in Figure 7 a. As seen in Figure 7 a, t_k^r are the antecedents, where r = 1...R, R is the number of rules and k = 1...a, a is the number of antecedents to be tuned, and t_{out}^r is the consequent of each rule, respectively.

In order to compute the cost function for BB-BC in this proposed technique, we use a confusion matrix to calculate an Average Recall. In our work the use of confusion matrix is convenient for a binary classifier (Fraud Transaction or Non-Fraud Transaction).

In binary classification, a Recall also called as sensitivity, or true positive rate, which is defined as the fraction of positive cases that were correctly identified [50], as follows:

$$\text{Recall Positive rate} = \frac{\text{TP}}{\text{TP} + \text{FN}} \quad (7)$$

Recall is calculated on the positive class and negative class by the formula:

$$\text{Recall Negative rate} = \frac{\text{TN}}{\text{TN} + \text{FP}} \quad (8)$$

Consequently, the average recall is:

$$\text{AVG } Recall \text{ rate} = \frac{\text{Recall positive} + \text{Recall Negative}}{2} \quad (9)$$

In a similar way of optimizing the rule base using BB-BC, the parameters of the MFs are encoded into a form of a population. The T2FLS MFs can be demonstrated as shown in Figure 7 b.

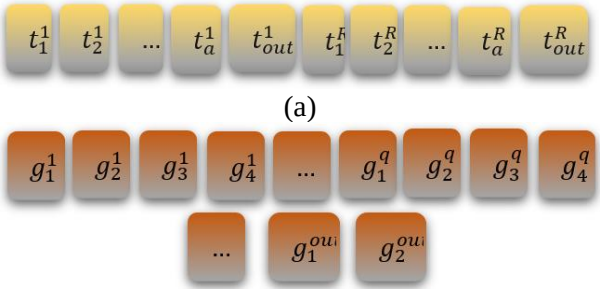

Figure 7: the population representation (a) parameters of rule base (b) parameters of T2 MFs

B. The input does not match any of the existing rules:

In [51] [52], to determine the output class for the input, in the case of incoming input vector x(p) does not match any of the existing X rules, once an input pattern is entered from the prediction dataset to the produced model, as mentioned before, we calculate the upper and lower membership values ($\bar{\mu}_{A_s^q}$, $\underline{\mu}_{A_s^q}$) for each inputs, and once the input matches many sets, then one input can generate more than one instance, and each rule will have an associated a firing strength, but not an output class. The next step is to find the closest rule in the rule base for each rule in MR ($x^{(p)}$), where MR ($x^{(p)}$) is the set of rules obtained by combining the matched fuzzy sets. To do this, we need to compute the similarity (or distance) versus each of the fuzzy rules produced.

Each rule in the rule base will have a similarity associated with the r-th rule generated from the input. For each rule in MR ($x^{(p)}$) the most similar rule is in the rule base, and by using above equation we can determine the output class. There will be "k" rules selected to decide for the $x^{(p)}$ input the output class (where "k" are the most similar rules to the k rules in MR ($x^{(p)}$)). Finally, the predicted class will be determined as a vote for each class; then the total vote strength can be computed by taking the highest $zClass_h$.

EXPERIMENTS AND RESULTS

This system used real-world financial dataset from Balad Bank, which consists of different types of interaction means, such as: ATMs, POS, mobile payment and internet Banking etc... The data was anonymized to remove personal information such

as Names, Mobile phones, Addresses, Card Number, etc.

Nevertheless, 17 significant features have been used for this research:

1. **Branch**: card holder's branch.
2. **Amount**: the amount of withdrawal (very small, small, mid or large amount).
3. **Gender**: male or female.
4. **Occupation**: card holder's job.
5. **Marital Status**: card holder's marital status (single, married, divorced or widowed).
6. **Time**: it displays when withdrawal was held (morning, day, night or mid night).
7. **Day of Month**: such as beginning of month, mid of month or end of month
8. **Week Day**: this feature presents the withdrawal week day (beginning, mid or end of Week).
9. **Age**: card holder's age such as (young, middle, old or very old).
10. **City**: city of terminal where withdrawal was held.
11. **Service Type**: it displays the type of service like: cash, bill payment, E15, NEC, or mobile top-up.
12. **Reference**: reference of withdrawal.
13. **Education**: education of the customer (none, basic, high school, B.Sc., PhD...).
14. **Account Type:** saving, current, investment or employee account.
15. **Withdrawal Type:** it displays the application or machine where the withdrawal was issued (mobile application, internet withdrawal, ATM/PoS withdrawal),
16. **Bank Terminal**: where withdrawal was held.
17. **Class**: fraud or non-Fraud.

The implementation of this experiment starting by data collection and then used FCM to extract the type-1 fuzzy sets from data, in addition to improve our result type-2 fuzzy set has been used while varying the FOU to 10%, 20% and 30%. After using the BB-BC, type-1 fuzzy sets and type-2 fuzzy sets utilized, respectively, calculating the fitness value with the selected rule base. In the training stage, 70% of this dataset has been chosen randomly, and the rest of this dataset was used for testing.

Designing Fuzzy sets using FCM:

FCM algorithm have been used to realise the type-1 fuzzy sets where Figure 8. Shows an example of the shapes the age fuzzy sets generated by FCM. We approximated the shapes shown in Figure 8 to generate convex normal type-1 fuzzy sets as shown in Figure 9.

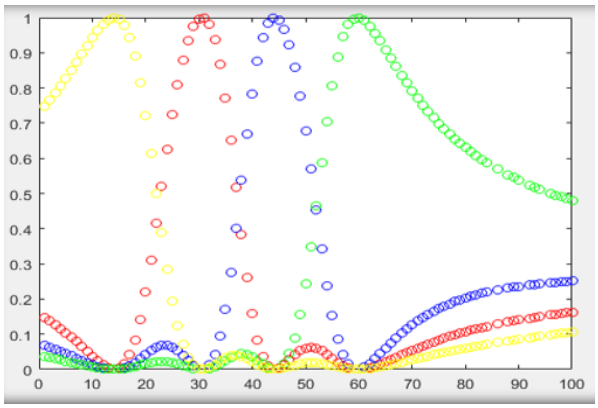

Figure 8: Type-1 Fuzzy Set Generated by FCM for Age

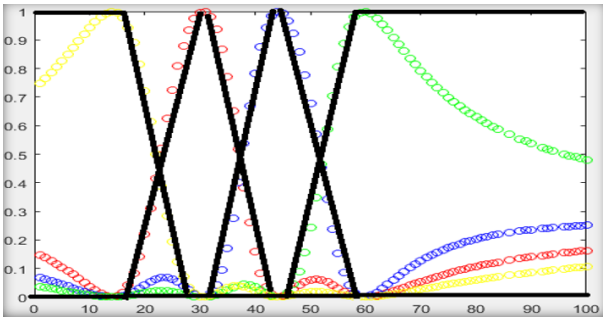

Figure 9: Generated Convex Normal Type-1 fuzzy sets from the FCM results in Figure 8.

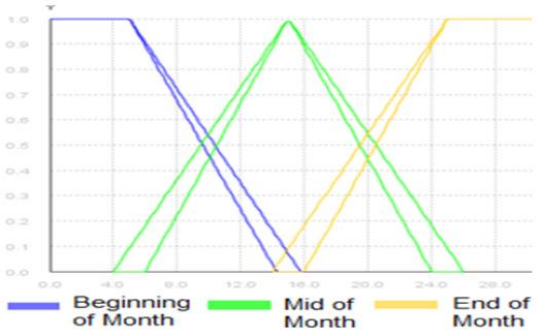

Figure 10: Type-2 Fuzzy Set for Day of Month Generated from Fuzzy Logic System

Our objective is not just to increase the accuracy in prediction, but to explain why the transaction is fraudulent or not. Nevertheless, we achieved a good result. To implement the BB-BC for Type-1 & Type-2 fuzzy logic system, JAVA programming language has been used. This application can be conFigured with any dataset and any number of

fuzzy sets The application has the ability to draw the fuzzy set as shown in Figure 10. It illustrates type-2 fuzzy set for Day of Month, with three sets (Beginning of month, mid of month and End of month). By using our previous experience, to compute the results for Type-1 fuzzy logic with equally spaced sets for prediction data and training data; then FCM used to generate type-1 fuzzy sets and we calculated the results for prediction data and training data again. The results for both were similar. That is to say 84% for prediction data, (as shown in Table 2.

Then type-2 fuzzy set applied with equal incremental in FOU for each fuzzy set as 10%, 20% and 30% for prediction data. Finally, BB-BC implemented to produce the best rule base, which was selected after receiving the maximum fitness, the BB-BC used with type-1 and type 2. The best result for type-2 was 87%, as shown in Table 3.

TABLE 2. AVG RECALL RATES FOR PREDICTING DATA IN TYPE-1 & TYPE-2 FLC USING FCM.

Type	Recall Positive	Recall Negative	AVG Recall
T1 (FCM)	99%	70%	84.5%
Type-2 FLC		**using FCM**	
10%	99%	66%	82.5%
20%	99%	72%	85.5%
30%	99%	72%	85.5%

TABLE 3. BEST RESULT BY USING BB-BC

Type	Recall Positive	Recall Negative	AVG Recall
Type-1	99%	65%	82%
Type-2	99%	74%	87%

In our model, each result can be simply read with clear justification, therefore the experts or the employees can easily read it. Also, the rule is very short and the rule base contains only 400 rules. Since the best AVG recall was 87%, two rules were selected from the rule base to be discussed; the first rule is shown in Figure 11 a:

From Figure 11 a, we can assume that someone used a worker's card with a very large amount at midnight, which is a suspicious behavior, because the worker has low income or salary thus he always uses a small or medium amount, and most of the worker's money is withdrawn daily for his everyday spending. This usually occurs in the morning, or at noon, not at midnight.

The example shown in Figure 11 b explains the simplicity of the rule. in this rule somebody used a youth's card, or actually a high school student's card to pay a customs duty, but as known the customs duty is a governmental payment for imported goods or cars, hence, a student cannot make such a transaction.

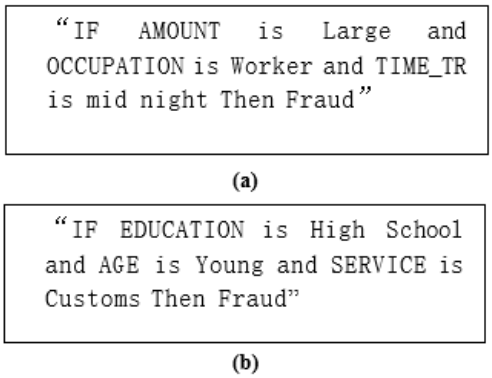

Figure 11: Actual Rules from the Model Rule Base (a) First Rule (b) Second Rule

Likewise, these rules were generated from the developed system, which are very significant and allow the financial sector in Sudan to track fraud patterns. Also, these rules are very simple, explainable and can be easily read. As we have realized from the above examples and from the learning phase, the developed system can generate rules that can deal with Sudanese society and can disclose any unacceptable behavior. Any financial institution can simply explain how the fraud has occurred by using these transparent rules.

CONCLUSIONS

In this paper, we have developed an optimization method that uses a very limited rule base by using Big Bang–Big Crunch optimization (BB–BC) approach with type-2 fuzzy logic system that can detect the fraud in Sudanese banks and financial sector, starting with any type of online transaction thru debit cards, and using real world dataset from the Balad Bank. Finally, this light rule base utilized for the prediction stage. We achieved 82% AVG recall for type-1 and 87% AVG recall for Type-2. Some examples of rules were generated by the proposed system and explained the simplicity of the rules, which can help to identify fraud patterns, and stop the occurrence of fraud in the financial sector in Sudan.

References

[1] The Nilson Report, https://www.prnewswire.com/news-releases/payment-card-fraud-losses-reach-27-85-billion-300963232.html

(retrieved 15 January 2020).
[2] Fraud The Facts 2019- The definitive overview of payment industry fraud, https://www.ukfinance.org.uk/system/files/Fraud%20The%20Facts%202019%20-%20FINAL%20ONLINE.pdf (retrieved 15 January 2020).
[3] Saeed, S. K. and Hagras, H., (2019). A Fraud-Detection Fuzzy Logic Based System for the Sudanese Financial Sector. *SUST Journal of Engineering and Computer Sciences (JECS)* **20**: 17-30.
[4] Erol, O. K., and Eksin, I. (2006). A new optimization method: big bang–big crunch. *Advances in Engineering Software* ***37*****(2):** 106-111.
[5] Qiu, S., Liu, Q., Zhou, S., and Wu, C. (2019). Review of artificial intelligence adversarial attack and defense technologies. *Applied Sciences* ***9*****(5)**: 909.
[6] Saeed, S. K., and Hagras, H. (2018). Adaptive Type-2 Fuzzy Logic Based System for Fraud Detection in Financial Applications. In: *Proceedings of the 2018 in 10th Computer Science and Electronic Engineering (CEEC).* Pp. 15-18. Colchester, United Kingdom. IEEE
[7] Oh, S. J., Schiele, B., & Fritz, M. (2019). Towards reverse-engineering black-box neural networks. In: *Proceedings of the Explainable AI: Interpreting, Explaining and Visualizing Deep Learning.* Pp. 121-144. Springer, Cham.
[8] Wang S.-C., (2003), Artificial neural network, *Interdisciplinary computing in java programming:* 81–100.
[9] Zareapoor M., Seeja K. R. and Alam M. A., (2012), Analysis on Credit Card Fraud Detection Techniques: Based on Certain Design Criteria, *International Journal of Computer Applications* **52**: 35-42.
[10] Zojaji Z., Atani R. E. and Monadjemi A. H., (2016), A Survey of Credit Card Fraud Detection Techniques: Data and Technique Oriented Perspective.
[11] Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., ... and Bengio, Y. (2014). Generative adversarial nets. In: *Proceedings of the Advances in neural information processing systems.* Pp. 2672-2680.
[12] Zheng, Y. J., Zhou, X. H., Sheng, W. G., Xue, Y., & Chen, S. Y. (2018). Generative adversarial network based telecom fraud detection at the receiving bank. In: *Proceedings of the Neural Networks*, *102*. Pp78-86.
[13] Cortes C. and Vapnik V., (1995), Support-vector networks, *Machine learning* **20**: 273–297.
[14] Kamboj M. and Gupta S., (2016), Credit Card Fraud Detection and False Alarms Reduction using Support Vector Machines, *International Journal of Advance Research, Ideas and Innovations in Technology,* **2**.
[15] Gyamfi, N. K., & Abdulai, J. D. (2018). Bank Fraud Detection Using Support Vector Machine. In *2018 IEEE 9th Annual Information Technology, Electronics and Mobile Communication Conference (IEMCON).* Pp. 37-41. IEEE.
[16] Srivastava v., Kundu A., Sural S. and Majumdar A., (2008), Credit card fraud detection using hidden Markov model, *IEEE Transactions on dependable and secure computing* **5**: 37–48.
[17] Gade V. and Chaudhari S., (2012), Credit card fraud detection using Hidden Markov Model, *International Journal of Emerging Technology and Advanced Engineering,* **2:** 511-513.
[18] Lucas, Y., Portier, P. E., Laporte, L., Calabretto, S., Caelen, O., He-Guelton, L., & Granitzer, M. (2019, April). Multiple perspectives HMM-based feature engineering for credit card fraud detection. In: *Proceedings of the 34th ACM/SIGAPP Symposium on Applied Computing.* Pp. 1359-1361.
[19] Delibašić B., Vukićević M., Jovanović M. and Suknović M. (2013), White-Box or Black-Box Decision Tree Algorithms: Which to Use in Education?, *IEEE Transactions on Education* **56**: 287–291.
[20] Shen A., Tong R. and Deng Y., (2007), Application of classification models on credit card fraud detection, In: *Proceedings of the 2007 IEEE International Conference on Service Systems and Service Management,* pp 1- 4, Chengdu, China.
[21] Hu H.-W., Chen Y.-L. and Tang K., (2009), A dynamic discretization approach for constructing decision trees with a continuous label, *IEEE Transactions on Knowledge and Data Engineering* **21**: 1505–1514.
[22] Save, P., Tiwarekar, P., Jain, K. N., & Mahyavanshi, N. (2017). A novel idea for credit card fraud detection using decision tree. *International Journal of Computer Applications*, **161(13):** 6-9.
[23] Sahin, Y., Bulkan, S., & Duman, E. (2013). A cost-sensitive decision tree approach for fraud detection. *Expert Systems with Applications*, **40(15)**: 5916-5923.
[24] Breiman, L. (2001). Random forests. *Machine learning*, **45(1)**: 5-32.
[25] Xuan, S., Liu, G., Li, Z., Zheng, L., Wang, S., & Jiang, C. (2018, March). Random forest for credit card fraud detection. In: *Proceedings of the 2018 IEEE 15th International Conference on Networking, Sensing and Control (ICNSC).* Pp. 1-6. IEEE.
[26] Aamodt A. and Plaza E., (1994), Case-based reasoning: Foundational issues, methodological variations, and system approaches, *AI communications* **7**: 39–59.
[27] Wheeler R. and Aitken S., (2000), Multiple algorithms for fraud detection, *Knowledge-Based Systems* **13**: 93–99.
[28] Malini, N., & Pushpa, M. (2017). Analysis of credit card fraud identification techniques based on KNN and outlier detection. In: *Proceedings of the 2017 Third International Conference on Advances in Electrical, Electronics, Information, Communication and Bio-Informatics (AEEICB).* Pp. 255-258. IEEE.
[29] del Mar Roldán-García, M., García-Nieto, J., &

Aldana-Montes, J. F. (2017). Enhancing semantic consistency in anti-fraud rule-based expert systems. *Expert Systems with Applications* **90**: 332-343.

[30] Kou, Y., Lu, C. T., Sirwongwattana, S., & Huang, Y. P. (2004). Survey of fraud detection techniques. In: *Proceedings of the 2004 IEEE International Conference on Networking, Sensing and Control* **2:** Pp. 749-754. Taipei, Taiwan.

[31] Leonard K. J., (1993), Detecting credit card fraud using expert systems, *Computers & industrial engineering* **25**:103–106.

[32] Bentley P. J., Kim J., Jung G.-H. and Choi J.-U., (2000), Fuzzy darwinian detection of credit card fraud, In: *Proceedings of the 14th Annual Fall Symposium of the Korean Information Processing Society*.

[33] Lee C.-C., (1990), Fuzzy logic in control systems: fuzzy logic controller. part II, *IEEE Transactions on systems, man, and cybernetics* **20**: 404–418.

[34] Hagras H., (2007), Embedding computational intelligence in pervasive spaces, *IEEE Pervasive Computing* **6**.

[35] Hagras H., Wagner C. (2009), Introduction to Interval Type-2 Fuzzy Logic Controllers - Towards Better Uncertainty Handling in Real World Applications, The IEEE Systems, an and Cybernetics eNewsletter, **27**.

[36] Mendel J. M., (2003), Type-2 fuzzy sets: some questions and answers, *IEEE Connections, Newsletter of the IEEE Neural Networks Society* **1**: 10–13.

[37] Hagras H., (2007), Type-2 FLCs: A new generation of fuzzy controllers, *IEEE Computational Intelligence Magazine* **2**: 30–43.

[38] Li H. L. L. and Lin W. H., (2006) Type-2 fuzzy logic approach for short-term traffic forecasting, In: *Proceedings IEEE transactions on Fuzzy Systems*, pp. 33–40.

[39] Erol, O. K., & Eksin, I. (2006). A new optimization method: big bang–big crunch. *Advances in Engineering Software*, **37(2)**: 106-111.

[40] Kumbasar, T., & Hagras, H. (2014). Big Bang–Big Crunch optimization based interval type-2 fuzzy PID cascade controller design strategy. *Information Sciences*, **282**: 277-295.

[41] Chimatapu, R., Hagras, H., Starkey, A., & Owusu, G. (2018, July). A Big-Bang Big-Crunch Type-2 Fuzzy Logic System for Generating InterpreTable Models in Workforce Optimization. In *Proceedings of the 2018 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE).* Pp. 1-8. IEEE.

[42] Camp, C. V. (2007). Design of space trusses using Big Bang–Big Crunch optimization. *Journal of Structural Engineering*, **133(7):** 999-1008.

[43] Kazemzadeh Azad, S., Hasançebi, O., & Erol, O. K. (2011). Evaluating efficiency of big-bang big-crunch algorithm in benchmark engineering optimization problems. *Iran University of Science & Technology*, **1(3):** 495-505.

[44] Camp, C.V. (2007). *Design of space trusses using big bang big crunch optimization*. Journal of Structural Engineering, **133(7)**: 999–1008.

[45] Kumbasar, T., Eksin, I., Guzelkaya, M., & Yesil, E. (2011). Type-2 fuzzy model inverse controller design based on BB-BC optimization method. *IFAC Proceedings Volumes*, **44(1)**: 5308-5313.

[46] Prayogo, D., Cheng, M. Y., Wu, Y. W., Herdany, A. A., & Prayogo, H. (2018). Differential Big Bang-Big Crunch algorithm for construction-engineering design optimization. *Automation in Construction*, **85:** 290-304.

[47] Yao, B., Hagras, H., Alghazzawi, D., & Alhaddad, M. J. (2016). A big bang–big crunch type-2 fuzzy logic system for machine-vision-based event detection and summarization in real-world ambient-assisted living. *IEEE Transactions on Fuzzy Systems*, **24(6)**: 1307-1319.

[48] N. R. Pal and J. C. Bezdek, (1995) ,On cluster validity for the fuzzy c-means model, *IEEE Transaction on Fuzzy Systems*, **3**: 370-379.

[49] Kesemen O., Tezel Ö. and Özkul E., (2016), Fuzzy c-means clustering algorithm for directional data (FCM4DD), *Expert Systems with Applications* **58:** 76–82.

[50] Fawcett T., (2006), An introduction to ROC analysis, *Pattern recognition letters* **27:** 861–874.

[51] Bernardo D., Hagras H. and Tsang E., (2013), A genetic type-2 fuzzy logic based system for the generation of summarised linguistic predictive models for financial applications, *Soft Computing A Fusion of Foundations, Methodologies and Applications* **17**: 2185–2201.

[52] Sanz J, Fernandez A, Bustince H, Herrera F, (2010), Improving the performance of fuzzy rule-based classification systems with intervalued fuzzy sets and genetic amplitude tuning. *Inf Sci* **180**:3674–3685.

[53] Andreu-Perez J., Cao F., Hagras H., (2018),Yang G., A self-adaptive online brain–machine interface of a humanoid robot through a general type-2 fuzzy inference system, IEEE Transactions on Fuzzy Systems, **26(1)**: 101-116, .

[54] Antonelli M., Bernardo D., Hagras H., and Marcelloni F., (2016), Multiobjective Evolutionary Optimization of Type-2 Fuzzy Rule-Based Systems for Financial Data Classification, *IEEE Trans. Fuzzy Syst.,* **25(2)**: 249-264.

[55] Hagras H., Colley M., Callaghan V., and Carr-west M., (2002), Online Learning and Adaptation of Autonomous Mobile Robots for Sustainable Agriculture, *Journal of Autonomous Robots*, **13**: 37–52.

[56] Starkey A., Hagras H., Shakya S., and Owusu G., (2016) A multi-objective genetic type-2 fuzzy logic based system for mobile field workforce area optimization, *Journal of Information Sciences*, **329**: 390–411,.

[57] Sakalli A., Kumbasar T., Yesil E., and H. Hagras, (2014), Analysis of the performances of type-1, self-

tuning type-1 and interval type-2 fuzzy PID controllers on the Magnetic Levitation system, In *Proceedings of the 2014 IEEE International Conference on Fuzzy Systems*.

[58] Lynch C., Hagras H., Callaghan V., (2006), Embedded Interval Type-2 Neuro-Fuzzy Speed Controller for Marine Diesel Engines", *In Proceedings of the International Conference on Information Processing and Management of Uncertainty in Knowledge-Based Systems*, Pp. 1340-1347, Paris, France.