Australian Journal of Basic and Applied Sciences, 7(8): 604-618, 2013
ISSN 1991-8178

# Adaptive Neuro Fuzzy Inference System and Artificial Neural Networks: reliable approaches for pipe stuck prediction

[1]Morteza Elahi Naraghi, [2]Peyman Ezzatyar, [3]Saeid Jamshidi

[1]University of Texas at Austin, Tx, USA.
[2]Islamic Azad University, Iran.
[3]Sharif University of Technology, Iran.

**Abstract:** In this paper, a novel approach for estimation of stuck pipe probability based on Artificial Intelligence has been proposed. Stuck pipe is one of the most prominent troubles occurring while drilling operations, and causes great time and money cost; thus, prediction of drilling pipe sticking can significantly reduce the cost of drilling, and enhance the efficacy. It is known that pipe sticking occurrence is a function of drilling operation parameters such as measured depth, Weight on Bit (WOB) and Revolution per Minute (RPM). Besides, it is believed that drilling fluid properties such as plastic viscosity, initial gel strength, 10 min gel strength and yield point can also affect pipe sticking. Recently, the prediction of stuck pipe while drilling via artificial intelligence, Artificial Neural Networks and Fuzzy Logic, is under investigation. In this paper, we examined capability of different types of static and dynamic neural networks such as feed forward back propagation network, feed forward time delay network, feed forward distributed time delay network, and layer recurrent network for pipe stuck prediction. Furthermore, a relatively novel method, i.e. Adaptive Neuro Fuzzy Inference System (ANFIS) is proposed for the same purpose. ANFIS is an architecture which is functionally equivalent to a Sugeno type fuzzy rule base. ANFIS is a method for tuning an existing rule base or membership functions with a learning algorithm based on a collection of training data. After the method is constructed and trained by gathered data, the method is tested by some other data in order to assess the method and evaluating the possibility of using the proposed method in industrial applications. As depicted later in this paper neurofuzzy classifiers can predict pipe sticking with 100% accuracy either in stuck data or none stuck data. Thus, the proposed methods possess reliable results for prediction of pipe stuck, and can be utilized in industrial softwares in order to prohibit the risk of pipe sticking.

**Key words:** Pipe stuck prediction, Artificial Neural Network, Adaptive Neuro Fuzzy Inference System, Feed-forward, Back-propagation.

## INTRODUCTION

Over several years oil industry is facing troubles associated with the stuck pipe. Differential pipe sticking is one of the stuck pipe mechanisms that have had a major impact on drilling efficiency and well costs (Adams, 1977a; Weakley, 1990; Wisnie and Zheiwei, 1994a). These occurrences are common everywhere in the world and are estimated to cost the industry hundreds of millions of dollars annually. In some areas, events related to differentially stuck pipe can be responsible for as much as 40% of the total well cost. Differential pipe sticking problems generally result in the significant amount of downtime and remedial costs and well cost overruns and time overruns as a non-productive time in terms of loss of rig days either due to stopping of drilling operations or an attempt to free the stuck pipe. This huge loss is always accounted in the well budget cost as a contingency factor for the risks associated with the stuck pipe problems in the well planning and drilling performance approach (Adams, 1997b; Beigler and Kuhn, 1994a; Wisnie and Zheiwei, 1994b; Sharif, 1997; Adnov *et al*., 1999). The recent increase in drilling activity, shortage of experienced personnel and equipment, and drilling in higher-risks areas have increased the risk of stuck pipe events in all drilling operations (Yarim *et al*., 2007).

The concept of differential pressure sticking of drill pipe was first reported by Helmick and Longley (Helmick and Longley, 1957) in according to laboratory tests. They stated that pipe sticking results when the drill pipe becomes motionless against a permeable bed and a portion of the area of the pipe is isolated by filter cake. In 1985, Kingsborough and Hemp king analyzed pipe sticking statistically based on drilling parameters (Hempkins *et al*., 1987). This was done by comparing the properties of non-pipe stuck wells with the ones which had stuck piping. Then drilling operations were planned according to non-pipe stuck wells' characteristics. In that study, 221 well's parameters were investigated in 131 stuck pipes' cases in Mexico's wells and the risk of stuck pipe occurrence in others wells were estimated. In 1994, Biegler and Kuhn generated a data based includes 22 drilling parameters in 73 non-pipe stuck wells and 54 pipe sticking wells in Mexico's gulf (Biegler and Kuhn, 1994b). These studies were the base of primary comparative analysis which could identify

**Corresponding Author:** Morteza Elahi Naraghi, University of Texas at Austin; Peyman Ezzatyar, Islamic Azad University; Saeid Jamshidi, Sharif University of Technology.

the pipe sticking mechanisms in addition to its probability prediction. In 1994, Glover and Howard improved the prediction stuck pipes' models by applying statistical techniques in 100 wells of Mexico's gulf (Howard and Glover, 1994).These models were used for prevention of pipe sticking and operation saving. Halliburton (Siruyuri *et al*., 2006) recently presented an application of Artificial Neural Network (ANN) methods for understanding the causes of differential stuck pipe.

Differential pipe sticking occurs when a part of the drill string, casing, or logging tool becomes embedded in a mud solids filter cake and is held there by a significant amount of differential pressure. This differential pressure is the pressure difference between the hydrostatic pressure of mud and the formation pore pressure. Usually, because of the excessive differential pressure, the sticking takes place across porous and permeable formations such as sandstone or limestone, where a mud filter cake builds up during drilling. It does not occur in very low permeability formations such as shales, where mud filter cakes normally do not form. Stuck pipe is identified as an impedance of drilling mud flow in the annular space and the difficulty of the pipe movement either in the upward or downward direction. In a complete stuck pipe situation, neither circulation nor pipe movement are possible.

Although these symptoms are similar to Key Seat sticking, they usually occur under different drilling conditions. Significant mud overbalance, as well as an exposed permeable section, must also exist for differential sticking to occur. It is clear that as many reservoirs become depleted, a significant number of wells will be drilled with high overbalance pressures, thereby maintaining the industry’s concerns over differential sticking (Miri *et al*., 2007).

The likelihood of differential sticking is increased further with the length of the permeable section that is open to the drilling fluid. The continued trend towards extended reach and horizontal drilling means that increasing lengths of permeable formations are exposed. Clearly, the nature of the rock formations encountered certainly cannot be altered. Therefore, if those formations carry a high risk of differential sticking, this has to be accepted. Also, high overbalance pressures may be unavoidable if they are needed to maintain well control or wellbore stability in other parts of the openhole section. However, mud composition and properties can be modified, within limits, in the prevention of differential sticking.

In the past multivariate statistical analysis techniques and simulated sticking testes using different drilling fluids have been performed to identify and modify parameters that lead to differential pipe sticking in order to prevent or minimize sticking. A review of published literature and laboratory data establishes the importance of mud filter cake properties (thickness, shear strength, and lubricity) on the differential sticking tendencies of mud.

As mentioned earlier, in this paper different tools of Artificial Intelligence, Adaptive Neuro Fuzzy Inference System and Artificial Neural Networks, are applied in order to predict the probability of pipe sticking. This paper is organized as follows. In the next part, a brief introduction of utilized materials which are Artificial Neural Networks and Neuro Fuzzy classifiers is presented. Then, within methodology part, overview of proposed method, and input and output parameters are discussed. Afterwards, the numerical implementation results are presented in results section.

### *Materials:*

A neural network is a robust tool for dynamic plant by means of a nonlinear regression in the discrete time domain. The result is a network, with adjusted weights, which can approximate the plant. One of the most enormous constraints is that the knowledge is stored in an opaque fashion; the learning results in a huge set of parameter values, almost infeasible to interpret in words. On the contrary, a fuzzy rule base consists of readable if-then statements that are similar to linguistic rules. In the fuzzy inference systems, the sample data is fuzzified. Fuzzification can be done by clustering data. Then, the rules are extracted. Afterwards, the output which is calculated based on the rules is deffuzified. These two methods can be combined in neurofuzzy systems in order to achieve the advantages of either of two methods, readability and learning ability, concurrently. The main idea of neurofuzzy systems is constructing a fuzzy inference system, and training the rules and modifying membership functions based on learning rules existing in neural networks. The obtained rules may disclose insight into the data that generated the model. In this paper, different types of Neural Networks in addition to Adaptive Neuro Fuzzy Inference Systems are utilized in order to predict pipe sticking. In the following part of this section, the applied materials which are Neural Networks and Neuro Fuzzy Systems are explained.

### *Artificial Neural Networks:*

Artificial Neural Networks (ANN) are enhanced by producing artificial neurons which are simple Processing Elements (PE) immensely interconnected in order to forge a small portion of the serial and parallel information processing capability of the biological neural networks. Several types of neural networks exist; each of them has specific sturdiness in order to become capable in particular usage. Their ability is germane to their structure, dynamics, and learning methods. The most prominent advantages of ANN is the ability of learning from examples, quick data processing, ease of insertion into existing and newly developed systems (Maen and Pap, 1990).

As mentioned above, in this paper a neural network-based pipe stuck prediction has been deliberated. In this method, the major parameters affecting pipe stuck mentioned above have been considered as input parameters of the method. These input parameters have been conducted to neural networks as the input layer, and the output of the network is the probability of pipe stuck. As discussed earlier, in this paper, a comparison between different types of neural networks, i.e. static type and dynamic type in performance of prediction of pipe sticking has been done. In the following section, a brief explanation on applied dynamic and static networks has been presented.

***Static and Dynamic Models:***

ANNs can be assorted into two main categories i.e. static and dynamic networks. Static networks possess no delay, and all connections are feed forwards, and the output is calculated directly via forward connections. However, in the dynamic networks, the output depends upon current inputs in addition to previous inputs and previous network states.

Dynamic networks are divided into two parts; those that have feedforward connections, and those that have feedback connections. Dynamic networks are generally more robust than static ones, and the training procedures are the same, yet training of dynamic networks are somewhat more difficult than those of static networks. The training algorithm is the same gradient-based algorithms used in static ones, but the performance of the algorithm in dynamic networks can be completely different, and the gradient should be computed in a more complex way. Thus, the dynamic back-propagation algorithm requires more time, and the error surface can be more complex. Training is trapped more often in local minima, and there may be more times required to train network to achieve the expected results.

***Network Architecture Design:***

As mentioned above, in this paper two different types of networks, static and dynamic, have been selected and used in order to evaluate the accuracy and performance of the method. The numbers of input and output neurons are imposed by those of input and output parameters which are respectively 7 and 1 in our case study. In either of the static and dynamic model, a network with one hidden layer has been proposed which is proved that is able to approximately model each continuous function (Fausett, 1993). The 10 hidden neurons have been considered for constructing a network which was empirically found to be the optimum number of hidden neurons. In order to evaluate the best network architecture suiting our problem, diverse architectures have been considered by which the training and testing steps have been implemented. The feedforward back propagation, with tangent sigmoid transfer function has been considered as static network. The Feed forward time delay and feed forward distributed time delay are opted as dynamic networks having only feedforward connections, and layer recurrent network has been selected as dynamic one having feedback connections. The schematics of applied networks are depicted in figures1-4. It should be considered that back-propagation algorithm was utilized in order to train all applied networks, and the activation functions of all layers are tangent sigmoid. In following parts of this section brief introduction to applied networks has been presented.

***Feed Forward Network:***

Figure 1 shows a typical two layer feedforward back-propagation network. As can be seen in figure 1, there is neither a feedback connection nor delay in networks, and the network is deemed as static one.

***Feed Forward Time Delay Network:***

Figure 2 exhibits a two layer feedforward time delay network. As can be seen in figure 2, the Feedforward time delay network is the most straightforward dynamic network consisting of a feedforward network with a Tapped Delay Line (TDL) at the input. In this network, the dynamics appear only at the input layer of a static multilayer feedforward network.

***Feed Forward Distributed Time Delay Network:***

Figure 3 depicts a two layer feedforward distributed time delay. As can be seen in figure 3, the TDLs are distributed throughout the network.

***Layer Recurrent Network:***

Figure 4 shows a two-layer Layer Recurrent Network (LRN). In this network, there is a feedback loop with a single delay, around each layer of the network except for the last layer.

***Back-Propagation Algorithm:***

As mentioned above, the back-propagation algorithm has been conducted in order to train all of applied ANNs. This algorithm encompasses following steps:

Step 1. Initialization i.e. assigning a small real random value to all biases and weights.

Step 2. Presentation of inputs and corresponding outputs. The inputs are the seven parameters discussed earlier, and the output is a vector determining corresponding individual.

Step 3. Calculating of outputs based on following equation:

$$y_i = f\left(\sum_{j=1}^{N-1} w_{ij}^{M-1} x_j^{M-1} + b_i^{M-1}\right), i = 1, \ldots N \tag{1}$$

Step 4. Weight and bias adaptation based upon following equations:

$$\Delta\omega_{ij}^{l}(k) = \mu x_j(k)\delta_i^l(k) \tag{2}$$

$$\Delta b_i^l(k) = \mu\delta_i^l(k) \tag{3}$$

where

$$\delta_i^l(k) = \begin{cases} \varphi^{'}(net_i^l)[d_i - y_i(k)] & l = M \\ \varphi^{'}(net_i^l)\sum_p \omega_{pi}\delta_p^{l+1}(k) & 1 \le l \le M-1 \end{cases} \tag{4}$$

And $x_j(k)$ is output of node $j$ at iteration $k$, $l$ is layer, $p$ is the number of output nodes of neural network, M is output layer, $\varphi$ is activation function. The learning rate is $\mu$. It should be considered that the higher learning rate is, the more quickly the convergence will be achieved. In this study, the learning rate is chosen as 1.0.

***Neuro Fuzzy Systems:***

In this section, basic principle of neuro fuzzy classifiers which is fuzzy logic is explained. Afterwards, neurofuzzy classifiers and Adaptive Neuro Fuzzy Systems are discussed.

***Fuzzy Logic Classifiers:***

As mentioned above, each fuzzy set consists of following steps. The first step is to fuzzify the data. Then, the rules should be extracted. The if-then rules are used to simulate any system, and compute the output. Then, the output which is computed based on the extracted rules is deffuzzified. In the fuzzification step, two approachesexist. It can be done by trial and error to find the most optimum membership functions. It can also be found by clustering the data. In the following part, the clustering method is described.

***Clustering:***

It can be rather difficult to fit the fuzzy model to the target data using trial and error, although it is quite easy to express linguistically the relation between input and output. A better approach is to approximate the target function with a piece-wise linear function and interpolate, in some way, between the linear regions.

In the Takagi-Sugeno model (Takagi and Sugeno, 1985)the idea is that each rule in a rule base defines a region for a model, which can be linear. The left-hand side of each rule defines a fuzzy validity region for the linear model on the right-hand side. The inference mechanism interpolates smoothly between each local model to provide a global model. The general Takagi-Sugeno rule structure is

If $f(e_1 \text{ is } \mathrm{A}_1, e_2 \text{ is } \mathrm{A}_2, ..., e_k \text{ is } \mathrm{A}_k)$ then $y = g(e_1, e_1, ...)$

Here *f* is a logical function that connects the sentences forming the condition, *y* is the output, and *g* is a function of the inputs $e_i$ .

***Introduction to Fuzzy Clusters (FCM Algorithm):***

It is reasonable to presume that points in the middle region between the two cluster centers have a gradual membership of both clusters. The fuzzified c-means algorithm (Jang *et al*., 1997) allows each data point to belong to a cluster to a degree specified by a membership grade, and thus each point may belong to several clusters.

The fuzzy c-means (FCM) algorithm partitions a collection of K data points specified by *m*-dimensional vectors, $u_k(k = 1, 2, ..., K)$ into *c* fuzzy clusters, and finds a cluster centre in each, minimizing an objective function. The most prominent point differentiating Fuzzy c-means from hard c-means is employing fuzzy partitioning in Fuzzy c-means i.e. a point can belong to several clusters with degrees of membership. To accommodate the fuzzy partitioning, the membership Matrix M is allowed to have elements in the range [0 1]. A point's total membership of all clusters, however, must always be equal to unity to maintain the properties of the M matrix. The objective function is defined as follows:

$$J(M, c_1, c_2, ..., c_c) = \sum_{i=1}^{c} J_i = \sum_{i=1}^{c} m_{ik}^q d_{ik}^2, \qquad (5)$$

where $m_{ik}$ is a membership between 0 and 1, $c_i$ is the centre of fuzzy cluster i, $d_{ik} = \|u_k - c_i\|$ is the first order Euclidean distance between the $i^{th}$ cluster centre and $k^{th}$ data point, and $q \in [1, \infty)$, is a weighting exponent. There are two requisite conditions which should be served in order to minimize *J*. The conditions are described as equation 6 and 7.

$$c_i = \frac{\sum_{k=1}^{K} m_{ik}^q u_k}{\sum_{k=1}^{K} m_{ik}^q} \qquad (6)$$

$$m_{ik} = \frac{1}{\sum_{j=1}^{c} \left( \frac{d_{ik}}{d_{jk}} \right)^{\frac{2}{q-1}}} \qquad (7)$$

The algorithm is based upon the preceding two conditions (Jang *et al*., 1997). The fuzzy c-means algorithm determines the cluster centers $c_i$ and the membership matrix M by pursuing the following steps:

1. Initialize the membership matrix M with random values between 0 and 1.
2. Calculate f cluster centers $c_i$ (i=1, 2,…, c).
3. Compute the objective function given the equation 1. If either it is below a certain threshold level or its improvement over the previous iteration is below a certain tolerance, the procedure ought to be stopped.
4. Compute a new M.
5. Go to step 2.

The cluster centers can alternatively be initialized first, before carrying out the iterative procedure. The algorithm may not converge to an optimum solution and the performance depends upon the initial cluster centers, just as in the case of the hard c-means algorithm.

***Neurofuzzy Function Approximation:***

It can be seen that the activation functions of a neural network look like fuzzy membership functions. In other words, the inference mechanism can be drawn in a block diagram somewhat like a neural network (Fig. 5). The network has an input layer, one hidden layer, and one output layer, and is germane to very simple fuzzy rule based. The input node connects to the neurons in the hidden layer. This node pertains to the if-part of the rules. Each neuron only encompasses an activation function. There is no summation, for each neuron has only one input. The one neuron in the output layer, with a rather odd appearance, calculates the weighted average corresponding to the centre of gravity defuzzification in the rule base. The network can be generalized to multi-input-multi-output control, but then the diagram becomes very busy.

Back propagation applies to this network since all layers are differentiable. Two possibilities for learning are apparent. One is to adjust the weights in the output layer, i.e., all the $w_i$ until the error is optimized. The other is adjusting the shape of the membership functions, provided that they are parametric.

The network can be described as a feed forward network with an input layer, a single hidden layer, and an output layer consisting of a single unit. The network performs a nonlinear mapping from the input layer to the hidden layer, followed by a linear mapping from the hidden layer to the output layer. Exactly, such a topology occurs in radial basis function networks; the hidden units provide a 'basis' for the input patterns and their functions 'radially' surround a particular data point. The details of radial basis networks are presented in Appendix A.

***Adaptive Neuro Fuzzy Inference System (ANFIS):***

ANFIS (Adaptive Neuro Fuzzy Inference System) is an architecture which is functionally equivalent to a Sugeno type fuzzy rule base (Jang and Sun, 1995; Jang *et al*., 1997). Under certain minor constraints the ANFIS architecture is also tantamount to a radial basis function network. Figure 6 depicts a typical ANFIS structure. ANFIS is a method for tuning an existing rule base or membership functions with a learning algorithm based on a collection of training data. This allows the rule base to adapt. The network in Fig. 6 may be extended by assigning a linear function to the output weight of each neuron,

$$w_k = a_k^T u + b_k, \qquad k = 1,2,...,K \tag{8}$$

where $a_k \in R^m$ is a parameter vector and $b_k$ is a scalar parameter. The network is then equivalent to a first order Sugeno type fuzzy rule base (Takagi and Sugeno, 1985). The requirements for the radial basis function network to be equivalent to a fuzzy rule base are summarized in the following (Jang *et al*., 1997).

- Both must use the same aggregation method (weighted average or weighted sum) to derive their overall outputs.
- The number of activation functions must be equal to the number of fuzzy if-then rules.
- When there are several inputs in the rule base, each activation function must be equal to a composite input membership function. One way to achieve this is to employ Gaussian membership functions with the same variance in the rule base, and apply product for the **and** operation. The multiplication of the Gaussian membership functions becomes a multi-dimensional Gaussian radial basis function.
- Corresponding activation functions and fuzzy rules should have the same functions on the output side of the neurons and rules respectively.

If the training data are contained in a small region of the input space, the centers of the neurons in the hidden layer can be concentrated within the region and sparsely cover the remaining area. Thus only a local model will be formed and if the test data lie outside the region, the performance of the network will be poor. On the other hand, if one distributes the basis function centers evenly throughout the input space, the number of neurons depends exponentially on the dimension of the input space.

***Methodology:***

In the previous section, basic materials used in this study were presented. As mentioned above, in this study a neurofuzzy system in addition to static and dynamic nueral networks are applied in order to predict the probability of pipe sticking. The materials utilized in constructing a neurofuzzy system are fuzzy inference system, and learning rules of networks. In this paper, Adaptive Neuro Fuzzy Inference Systems are used as neurofuzzy classifier. In next sections, data gathering, data preprocessing and evaluation method are explained.

***Data Gathering:***

As mentioned above, the most prominent goal of this study is to construct an inference system by which the probability of pipe sticking can be estimated. The inference system possesses some inputs which are parameters of drilling operations through which the output which is pipe stuck probability is determined. The inputs are measured depth, plastic viscosity, yield point, initial gel strength, 10 min gel strength, Weight On Bit (WOB), and Revolution Per Minute (RPM). The block diagram of the system is depicted in Figure 2. In this section, the input parameters will be reviewed to provide an understanding of how they affect the potential of pipe sticking occurrence and the necessary force to pull the pipe free after it is stuck.

***Mud Type:***

A comparison of generic mud types has shown oil-based mud to have the lowest stickance values and gel-water based mud has the highest. Polymer-water-based mud fall between these two extremes. It was found that the sticking potential also varies greatly within a mud type, depending on the precise formulation tested (Reid *et al*., 2000).

***Lubricant:***

The addition of certain lubricants for water- and oil-based muds will reduce the effect of differential sticking. If sticking still occurs, then reduce the force needed to free the stuck pipe or tool.

***Solids Level:***

Type and amount of solids play a role in cake characteristics and affect the degree of pipe sticking and pull out force to get it free (Isambourg *et al*., 1999). Increasing the solids level in the mud (both weighting agent and drilled solids) has been found to increase the force needed to free the pipe. This effect depends on the type of mud used. For example, salt muds have the lowest sticking tendency until reactive drill solids are added, which resulted in one of the highest measured forces (Bushnell-Watson and Panesar, 1991).

***Fluid Loss:***

Improving fluid loss can reduce the stickance tendencies of a mud. Oil-based muds usually have low fluid loss values. However, reducing the fluid loss does not have the same effect on stickance in all mud systems (Isambourg *et al*., 1999). It is currently not possible to determine accurately the sticking potential of the mud from a single mud property, such as density, fluid loss, solids content, or lubricity. However, laboratory work has shown that several mud treatment options, including adding a lubricant, can reduce the sticking tendencies of a mud.

***Mud Cake Properties:***

A mud cake is formed on permeable formations if the formation pressure is significantly lower than the hydrostatic pressure of the drilling fluid. As a result, there is an invasion of the liquid phase into the permeable zone and deposition and/or penetration of the corresponding solids inside and against the formation (Courteille and Zurdo, 1985). After a period of time, equilibrium is reached and deposition is balanced by erosion, resulting in a constant cake thickness. However, when the mud is static, erosion then occurs and the cake thickness increases with time. Increasing the time that the mud is not circulating will increase mud cake thickness and the likelihood of differential sticking.

Cake thickness cannot be used alone to predict the sticking tendency of different types of mud. However, for one particular mud formulation, an increase in cake thickness will increase the force required to free the pipe. Darcy's Law predicts that the cake thickness will increase with the square root of time and the increase in the force to free the pipe also follows the same relationship. This suggests that the change in the sticking tendency is a result of increasing contact area. In general, to reduce the chance of differential sticking, the time that the mud is left static in the hole should be minimized (Bushnell-Watson and Panesar, 1991).

Another parameter that influences pipe sticking is the friction factor. The friction between steel and mud cake varies with changes in mud composition. Previous studies have shown that the friction factor increased with increased barite content of the mud. Carboxymethylcellulose had no effect on the friction factor. Emulsification of oil in the mud had the effect of reducing the friction factor. In summary, the mud composition may be altered to reduce the friction between the pipe and mud cake (Annis and Monaghan, 1962).

Several characteristics of the mud cake have an effect on pipe sticking and on the necessary force to pull out the pipe. The sticking tendency of a mud cake depends on more than one parameter. It will vary as a result of cake thickness (contact area) and mud cake properties (friction/adhesion and surface roughness). The combination of these factors means that predicting the sticking tendency of any mud is not simple (Murillo *et al*., 2009)

In this research, after analyzing the general properties of well and drilling fluid, seven most effective ones were selected to be used as input variables in the neural network model; these parameters are defined as follow:

***Measured Depth:***

The length of the wellbore is one of the parameters affecting pipe sticking. At greater depth, more stresses will be imposed on formation and it could be a major stuck pipe variable.

***Yield Point (YP):***

YP is the yield stress extrapolated to a shear rate of zero. YP is used to evaluate the ability of a mud to lift cuttings out of the annulus. A high YP implies a non-Newtonian fluid, one that carries cuttings better than a fluid of similar density but lower YP. YP is lowered by adding deflocculant to a clay-based mud and increased by adding freshly dispersed clay or a flocculant, such as lime.

***Plastic Viscosity (PV):***

PV is the slope of the shear stress/shear rate line above the yield point. A low PV indicates that the mud is capable of drilling rapidly because of the low viscosity of mud exiting at the bit. High PV is caused by a viscous base fluid and by excess colloidal solids. To lower PV, a reduction in solids content can be achieved by dilution of the mud.

***Gel Strength (Initial and 10-mins):***

The shear stress measured at low shear rate after a mud has set quiescently for a period of time (10 seconds and 10 minutes in the standard API procedure). Some drilling fluids are thixotropic, forming gelled structures when stagnant and liquefying when sheared. The specific gel strength of a drilling fluid is described as low-flat (most desirable), progressive or high-flat (both undesirable) according to its measured gel strength versus time.

***Weight on Bit (WOB):***

Weight on bit is an essential factor in the drilling process, which can affect the rate of penetration as well as natural frequencies of the drill string in the bending mode of vibration. The WOB can also be related to the load carrying capacity of the drill string (buckling load). Increasing the weight on bit will bend the drill collars behind the near-bit stabilizer more, so the rate of build will increase.

***Revolutions per Minute (RPM):***

A higher rotary speed will tend to `straighten' the drill collars and hence reduce the rate of build. Increasing the RPM of the bit provides more opportunities to cut the formation in a given amount of time.

***Data Processing:***

As mentioned earlier, the main goal followed in this paper is to construct and evaluate systems by which the probability of pipe sticking can be estimated, and the data can be classified in two parts which are stuck data and none stuck data. Consequently, our problem is considered as a classification problem. As all classification problems, the first step is to gather data, and preprocess them. In classification problems, the data is dividedinto two parts which are train data and test data. Train data is utilized to construct and train the system. The test data, which is independent of train data, is used to evaluate the method. The details of assessing the model by test data are discussed in next sections. In our study, 245 daily drilling reports of Doroud field are utilized to train and test the method. 194 of them are used to construct and train the model, and 51 of them are utilized in order to assess the method. At first, the used data is normalized based on following equation in order that all variables are in the interval [0 1].

$$Xn = \frac{X - X_{\min}}{X_{\max} - X_{\min}} \tag{9}$$

where $Xn$, $X$, $X_{\min}$, and $X_{\max}$ are normalized parameter, original parameter, minimum used parameter, and maximum used parameter.

***Evaluating Method and Error Calculation:***

The next step, after constructing the model, is to assess the model, and calculate the error. As mentioned earlier, as a classification problem, the data is divided in two parts. The first part which is train data has been used to construct the ANFIS or the different networks, and train them. The test data which is independent of train data is utilized in order to assess the method. The details are as follows. The test data is conducted to the model, and the outputs are calculated based on the constructed model. The next step is to calculate the error.

As discussed earlier, the main target followed by this study is to predict the pipe sticking while drilling. The data can be divided into two parts. Stuck data which is the data that pipe sticking has occurred while drilling, and none stuck data which is the data that no pipe sticking has occurred while drilling operation. In this study, the output of stuck data is assumed 1, and the output of none stuck data is assumed 0. The most important target of this study is to construct a model to determine whether pipe sticking will occur while drilling operation or not. Thus, our study is to classify the data into two parts; stuck data and none stuck data.

In order to calculate the error, the test data is conducted to trained model, and the output is calculated. The output is a real number in [0 1] interval. If the output is greater than 0.5 the data is considered as stuck data. Otherwise, the data is considered as none stuck data. Then, the error based on the comparison of actual output and determined one is calculated. The error is defined as follows.

$$\text{error}=\frac{\text{The number of stuck data considered as none stuck+The number of none stuck data considered as stuck}}{\text{total number of test data}}$$

***Results:***

Our purpose is to develop a system by which the probability of pipe sticking can be determined. The block diagram of the proposed method is illustrated in Figure 7. The input data which is discussed in introduction part are conducted to the inference system, and the output is the probability of pipe sticking. If the output is greater than 0.5 the case is considered as stuck, and if the output is less than 0.5 the case is considered as none stuck. In another word, our purpose is to classify the different sets of inputs as stuck or none stuck. As a classification issue, the data should be divided into two parts. The first one which is train data is used to train the inference system based on the known set of input and output data. The second part of data is test one which is utilized in order to determine the accuracy of the constructed inference system. Within the following part of this section, the training and testing process is explained.

After the data is processed based on data processing section, the normalized train data is used in order to construct the inference system. This study can be divided into two parts. The first part is the prediction based on ANNs, and the second is the prediction based on ANFIS.

In the first section which is neural network-based classification issue data is divided into three categories; train data, validation data, and test data. Train data is used to train networks and compute the optimum biases and weights. The validation data is used in order to check the computed the network performance while training and stop the training process whenever the results are satisfactory. The test data is to assess the constructed network's performance.

In this case study, the 171 data of daily reports of drilling has been used for train and 37 data are used as validation data. All networks are trained based on the back-propagation algorithm which is a gradient-based algorithm for updating weights and biases. As discussed earlier, four types of networks which are feedforward, feedforward time delay, feedforward distributed time delay, and layer recurrent are used in order to predict pipe sticking. After data assembled according to process explained in the database assembly, the mentioned networks are constructed based on the explanations on network architecture design. Then, the back-propagation algorithm is applied on all the constructed networks. The performance of each network while training is depicted in figures 8-11.

In the second section which is ANFIS-based classifier, at first a fuzzy inference system ought to be created. The number and shape of membership functions for each parameter is respectively considered as 4 and Gaussian. In order to construct the inference system, FCM algorithm was implemented on the normalized train data. Then, the next step of fuzzy inference system construction which is fuzzification has been done, and fuzzy rules were extracted.

The next step is to design ANFIS system. The structure of the system has been designed as discussed in Appendix B. Then, the system should be trained. The learning rule which is mentioned in Appendix B has been conducted on the designed system. Figure 12 shows the membership function of one of the inputs before training. It should be considered that the membership function before training is determined based on FCM clustering algorithm. Figure 13 depicts the membership function after training by ANFIS learning rule.

Afterwards, the designed and trained ANFIS system must be assessed. As mentioned above, the evaluation of the constructed system should be done via normalized test data. The procedure is as follows. The inputs of the normalized test data should be conducted to the constructed system, and the predicted output is compared to the known one. As mentioned above, the output of the system is the probability of pipe sticking, and any probability greater than 0.5 is considered as stuck, and others are considered as none stuck.

Error has been calculated based on previous explanations, and the results of prediction of pipe sticking via different methods, static and dynamic networks and ANFIS, are exhibited in Table I. As can be seen, both ANN and ANFIS can predict pipe sticking with no error. Furthermore, among different types of ANNs, the performance of feedforward distributed time delay which is a dynamic network is the best ANN for prediction of pipe sticking.

***Discussion and Conclusion:***

In this paper, we examined ability of artificial intelligence methods for pipe stuck prediction. Stuck pipe is one of the most prominent troubles occurring while drilling operations, and causes great time and money cost; thus, prediction of drilling pipe sticking can significantly reduce the cost of drilling, and enhance the efficacy. In this paper, different types of dynamic and static neural networks such as feed forward back propagation network, feed forward time delay network, feed forward distributed time delay network and layer recurrent network are applied for the same purpose. . In addition, a novel approach for prediction of drilling pipe sticking based on estimation of probability of stuck pipe based on ANFIS is proposed. The proposed system is a Multiple Input Single Output system in which the output is the probability of stuck pipe, and the inputs are some of the mud and drilling operations characteristics such as measured depth, plastic viscosity, yield point, initial gel strength, 10 min gel strength, Weight On Bit (WOB), and Rotary Speed (RPM). As it can be seen in results section, both ANFIS and neural networks can estimate pipe sticking with a high accuracy, and among different types of neural networks, Feedforward distributed time delay structure which is a dynamic one is the best predictor network. Therefore, the proposed methods can be used in industrial softwares in order to minimize the risk of stuck pipe. Considering results section, both ANN and ANFIS can predict pipe sticking with approximately same accuracy. ANN possesses more simplicity than ANFIS, yet ANFIS can be utilized in more complex situations.

## REFERENCES

Aadnoy, B.S., K. Larsen, P.C. Berg, 1999. “Analysis of Stuck Pipe in Deviated Boreholes”, paper SPE 56628 presented at the SPE Annual Technical Conference and Exhibition, Texas, Oct.

Adams, N., 1977. “A Field Case Study of Differential Pressure Sticking”, paper SPE 6716 presented at the SPE Annual Technical Conference and Exhibition, Denver, Oct.

Annis, M.R., P.H. Monaghan, 1962. “Differential Pressure Sticking – Laboratory Studies of Friction Between Steel and Mud Filter Cake”,Journal of Petroleum Technology, 14(5).

Biegler, M.W., G.R. Kuhn, 1994. “Advances in Prediction of Stuck Pipe Using Multivariate Statistical Analysis”, Presented at the SPE/IADC Conference, Dallas, Texas, February.

Bushnell-Watson, Y.M., S.S. Panesar, 1991. “Differential Sticking Laboratory Tests Can Improve Mud Design”, Paper SPE 22549-MS presented at the SPE Annual Technical Conference and Exhibition, Dallas, Texas, October.

Courteille, J.M., C. Zurdo, 1985. "A New Approach to Differential Sticking", Paper SPE 14244 presented at the SPE Annual Technical Conference and Exhibition, Las Vegas, Nevada, September.

Fausett, L., 1993. "Fundamentals of Neural Networks: Architectures, Algorithms And Applications", 1st edition, Prentice Hall.

Helmick, W.R., A.J. Longley, 1957. "Pressure Differential Sticking of Drill Pipe and How It Can Be Avoided or Relieved", conference of Drilling and Production Practice.

Hempkins, W.B., R.H. Kingsborough, W.E. Lohec, C.J. Nini, 1987. "Multivariate Statistical Analysis of Stuck drill Pipe Situations", Journal of SPE Drilling Engineering, September.

Howard, J.A., S.S. Glover, 1994. "Tracking Stuck Pipe Probability While Drilling" paper SPE 27528 presented for Presentation at the 1994 IADC/SPE Drilling conference, Dallas, Texas, February.

Isambourg, P., S. Ottesen, S. Benaissa, J. Marti, 1999. "Down-Hole Simulation Cell for Measurement of Lubricity and Differential Pressure", Paper SPE 52816-MS presented at the SPE/IADC Drilling Conference, Amsterdam, Netherlands, March.

Jang, J.S.R., C.T. Sun, 1995. "Neuro-fuzzy modeling and control", Proceeding of the IEEE, 83(3).

Jang, J.S.R., C.T. Sun, E. Mizutani, 1997. "Neuro-Fuzzy and Soft Computing", number isbn 0-13-261066-3 in Matlab Curriculum Series, Prentice Hall, Upper Saddle River, NJ, USA.

Maen, A.J., C.T. Harston, R.M. Pap, 1990. "Handbook of Neural computing applications", Academic press Inc., San diego, California.

Miri, R., J. Sampaio, M. Afshar, A. Lourenco, 2007. "Development Of Artificial Neural Networks To Predict Differential Pipe Sticking in Iranian Offshore Oil Fields" paper SPE 108500 presented at the international Oil conference and Exhibition,Veracruz, Mexico, June.

Murillo, A., J. Neuman, R. Samuel, 2009. "Pipe Sticking Prediction and Avoidance Using Adaptive Fuzzy Logic and Neural Network Modeling" Paper SPE 120128 presented at Production and Operations Symposium, Oklahoma City, Oklahoma, USA, April.

Reid, P.I., G.H. Meeten, P.W. Way, B.D. Chambers, M.W. Sanders, 2000. "Differential–Sticking Mechanisms and a Simple Wellsite Test for Monitoring and optimizing Drilling Mud Properties", SPE Drilling & Completion journal, 15(2).

Sharif, Q.J., 1997. "A Case Study of Stuck Drillpipe Problems and Development of Statistical Models to Predict the Probability of Getting Stuck and If Stuck, the Probability of Getting Free", University of Engineering & Tech, Lahore, Pakistan, May.

Siruvuri, C., S. Nagarakanti, R. Samuel, 2006. "Stuck Pipe Prediction and Avoidance: A Convolutional Neural Network Approach", paper SPE 98378 presented at the 2006 IADC/SPE Drilling Conference, Miami, Florida, February.

Takagi, T., M. Sugeno, 1985. "Fuzzy identification of systems and its applications to modeling and control", IEEE Trans.Systems, Man & Cybernetics, 15(1).

Weakley, R.R., 1990. "Use of Stuck Pipe Statistics to Reduce the Occurrence of StuckPipe", SPE 20410.

Wisnie, A.P., Z. Zheiwei, 1994. "QuantifyingStuck Pipe Risk in Gulf of Mexico Oil and Gas Drilling", SPE 28298.

Yarim, G., R. Uchytil, R. May, A. Trejo, P. Church, 2007. "Stuck Pipe Prevention – A Proactive Solution to an Old Problem", Paper SPE 109914 presented at the SPE Annual Technical Conference and Exhibition, Anaheim, California, November.

***Appendix A. Radial Basis Networks***

Radial basis function networks are used for curve-fitting in a multi-dimensional space. This is also called function approximation, and learning is equivalent to finding a function that best fits the training data. In its strict sense the function is constrained to pass through all the training data points. The radial basis functions technique consists of choosing a function F,

$$\mathrm{F}(u) \quad = \quad \mathrm{w}^T \ \mathrm{f}\left(\left\|u\text{-}u_{\mathrm{k}}\right\|\right) \tag{1}$$

$$= \begin{bmatrix} w_1 & w_2 & \ldots & w_k \end{bmatrix} \begin{bmatrix} \mathrm{f}(\|\mathbf{u}\text{-}\mathbf{u}_1\|) \\ \mathrm{f}(\|\mathbf{u}\text{-}\mathbf{u}_2\|) \\ \cdots \\ \mathrm{f}(\|\mathbf{u}\text{-}\mathbf{u}_{\mathrm{K}}\|) \end{bmatrix}, \tag{2}$$

Here $u \in \mathrm{R}^{\mathrm{m}}$ is a vector of inputs, $u_k \in \mathrm{R}^{\mathrm{m}}$ (k=1,2,…,K) are vectors of training data, $w \in \mathrm{R}^{\mathrm{K}}$ is the vector of weights, $\mathrm{f}(\|\mathrm{u}\text{-}\mathrm{u}_{\mathrm{K}}\|)$ is a set of (nonlinear) radial basis functions. The known data points $u_{\mathrm{k}}$ are taken to be the centers of the radial basis functions. The activation level of a function $f(u, u_k)$ is maximum when the input u is at the centre $u_k$ of the function. Each known data point $u_k$ must gratify the equation 3,

$$w^T f(u_k) = d_k \tag{3}$$

Where $d_k$ is the desired response corresponding to $u_k$, therefore the unknown weights *w* must satisfy the following set of equations

$$w^T \Phi = d^T \tag{4}$$

Where the vector *d* represents the desired response vector, and the matrix $\Phi = \{f_{j,k}\}$ is called the interpolation matrix and $f_{jk} = f\left(\left\|u_j - u_k\right\|\right),\ j,k = 1,2,...,K$.

For a class of radial basis functions, Gaussian functions for instance, the interpolation matrix is invertible (it is positive definite). Provided the data points are all distinct, then the weights can be calculated directly by obtaining the following equation:

$$w^T = d^T \Phi^{-1} \tag{5}$$

However, it cannot practically used if the interpolation matrix is close to singular.

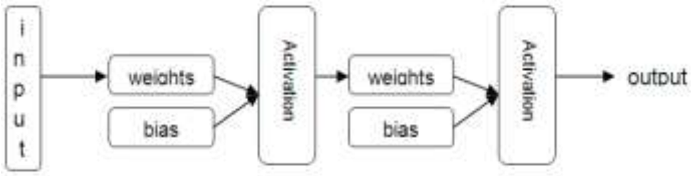



**Fig. 1:** A two layer feedforward back-propagation network structure.

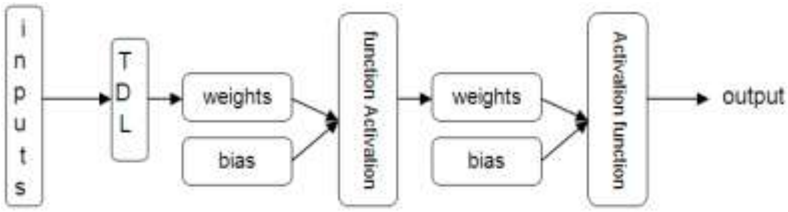



**Fig. 2:** A two layer feedforward time delay network structure.

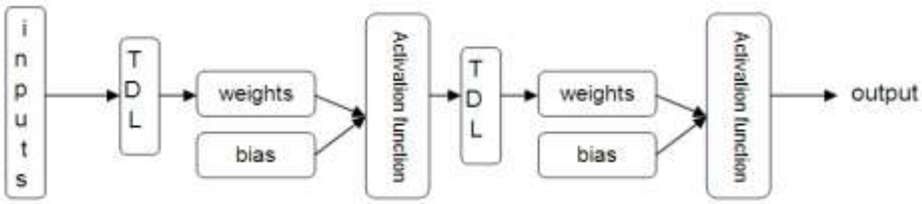



**Fig. 3:** A two layer feedforward distributed time delay network structure.

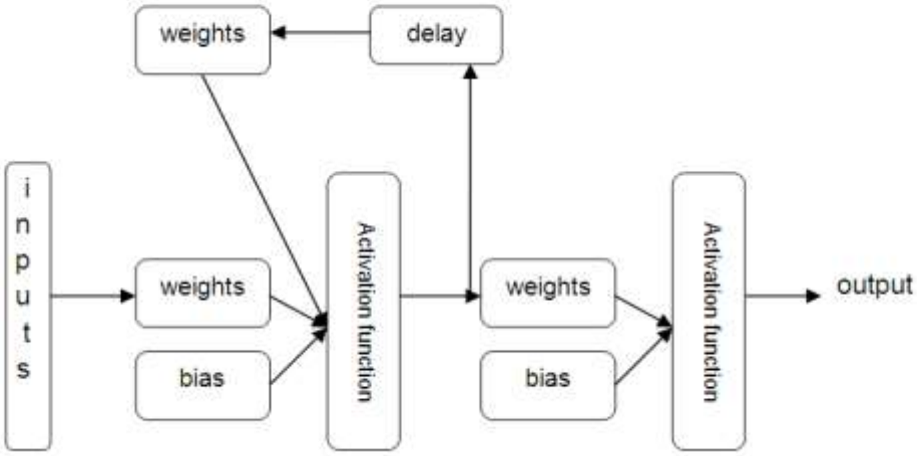



**Fig. 4:** A two layerLRN structure.

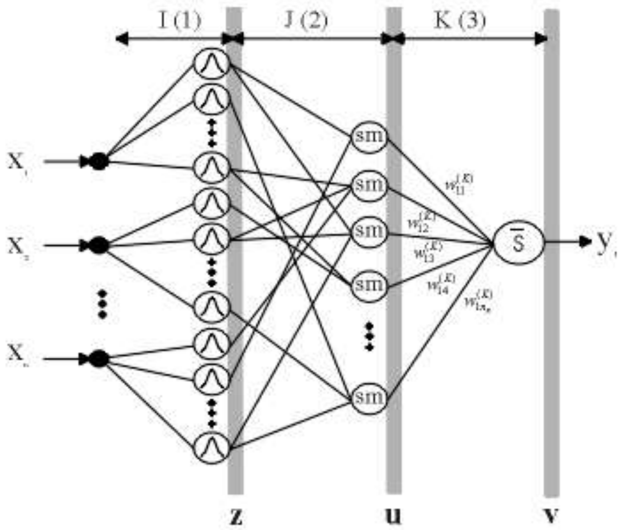
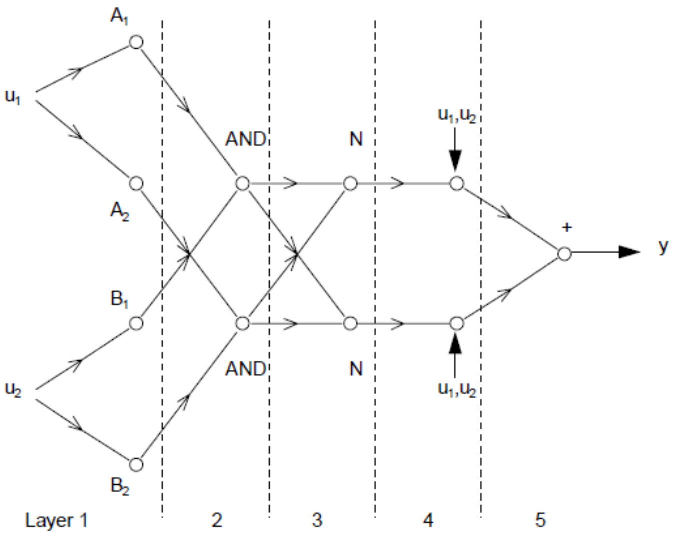



**Fig. 5:** Fuzzy rules perceived as a network.




**Fig, 6:** Structure of the ANFIS network.

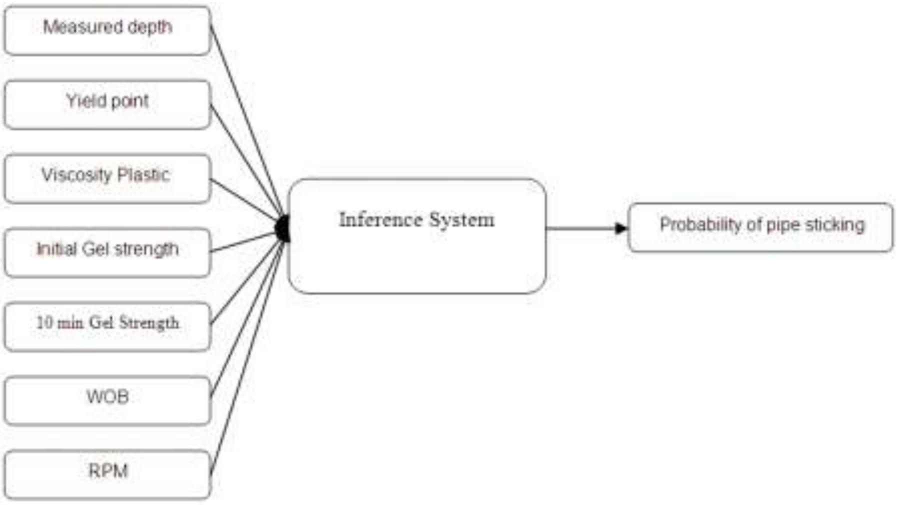



**Fig. 7:** The block diagram of the proposed method.

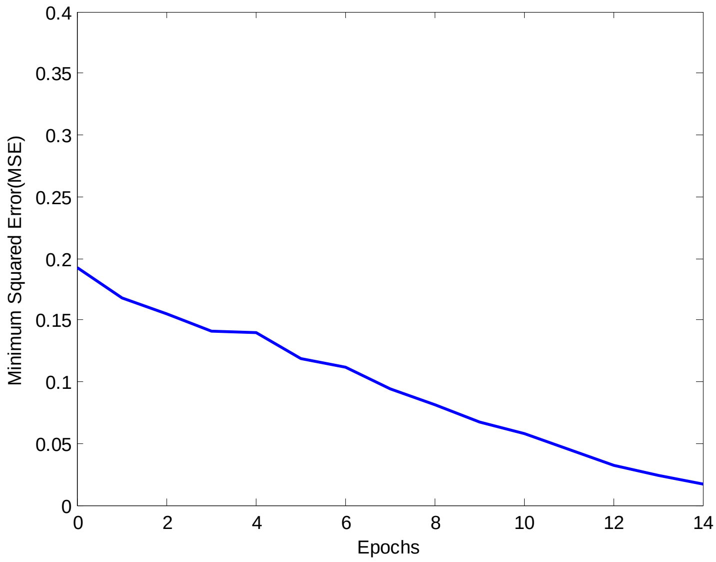



**Fig. 8:** The performance of feedforward network while training.

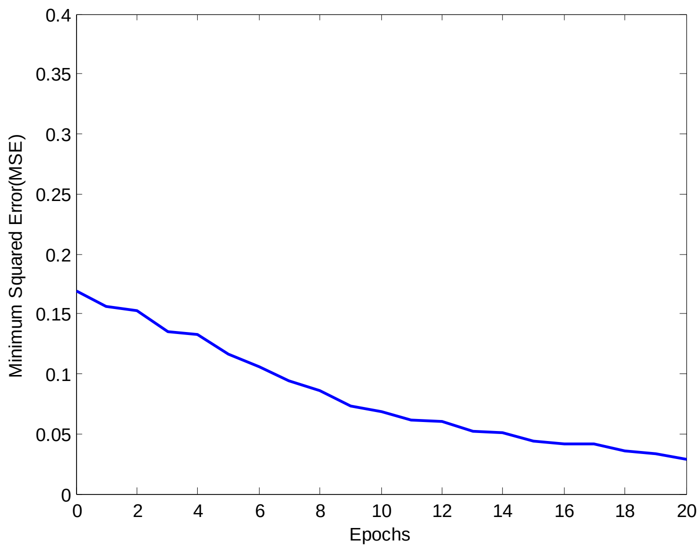



**Fig. 9:** The performance of feedforward time delay network while training.

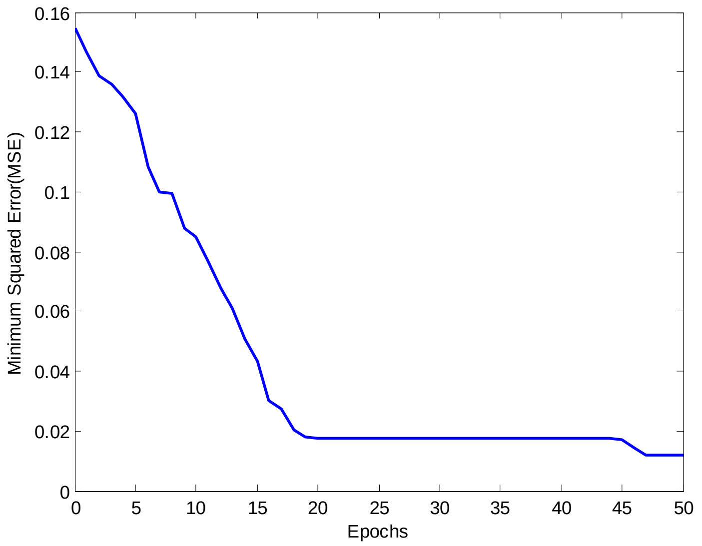



**Fig. 10:** The performance of feedforward distributed time delay network while training.

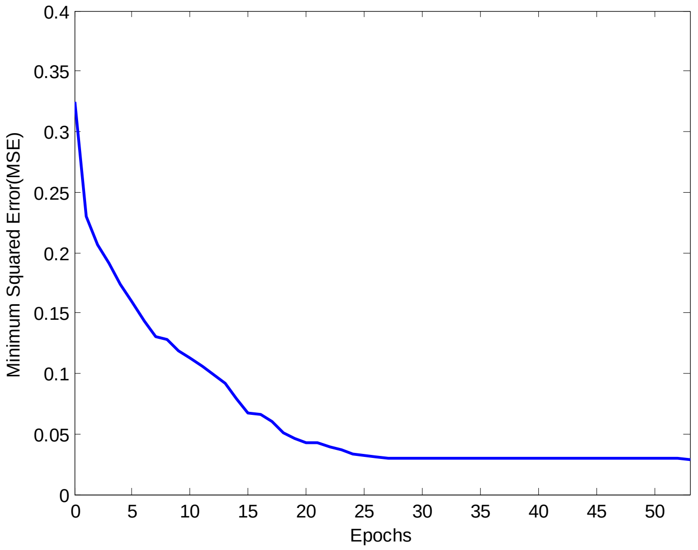



**Fig. 11:** The performance of layer recurrent network while training.

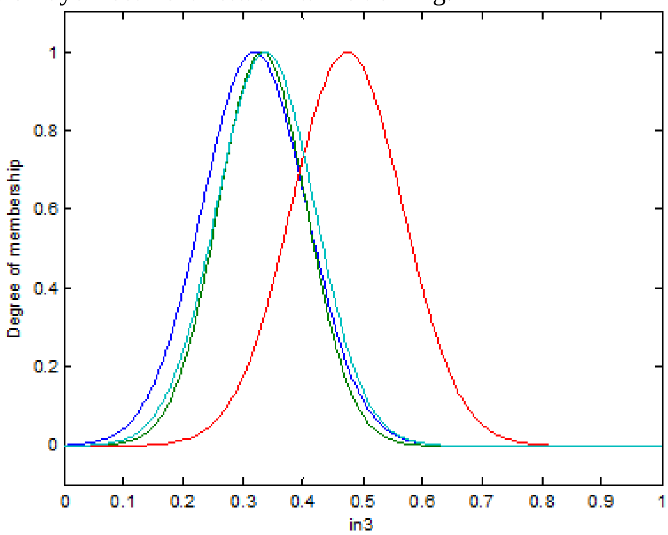



**Fig. 12:** The membership function of one of inputs (plastic viscosity) before training.

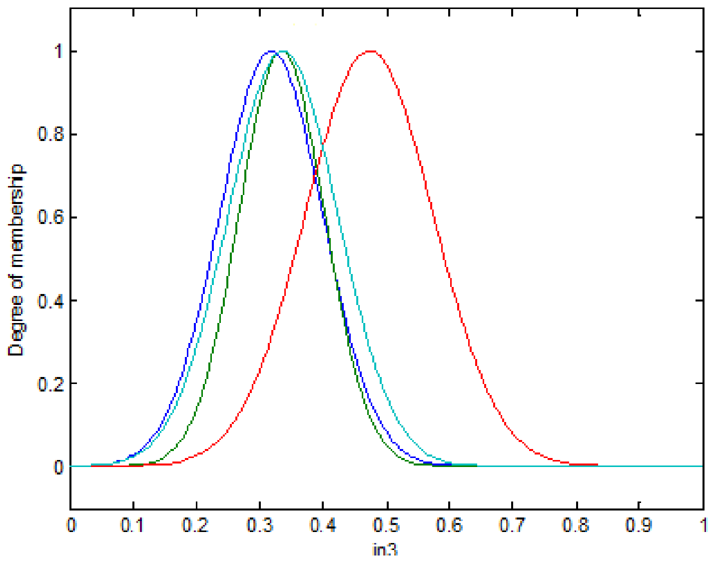



**Fig. 13:** The membership function of the input (plastic viscosity) after training.

**Table I:** The prediction accuracy of each method.

| Method | Error (%) |
| --- | --- |
| Layer recurrent network | 13 |
| Feedforward time delay network | 16 |
| Feedforward distributed time delay network | 0 |
| Feedforward back propagation network | 9 |
| Adaptive Neuro Fuzzy Inference System | 0 |