Predicting injection profiles using ANFIS

Mingzhen Wei a,*, Baojun Bai a, Andrew H. Sung b, Qingzhong Liu b, Jiachun Wang c, Martha E. Cather d

a University of Missouri-Rolla, 129 McNutt Hall, 1870 Miner Circle, Rolla, MO 65409, United States
b New Mexico Institute of Mining and Technology, 801 Leroy Place, Socorro, NM 87801, United States
c Daqing Petroleum Company Limited, PetroChina, Daqing, Haerbing, PR China
d New Mexico Petroleum Recovery Research Center/New Mexico Tech, 801 Leroy Place, Socorro, NM 87801, United States

Received 21 September 2005; received in revised form 16 March 2007; accepted 17 March 2007

Abstract

Decision making pertaining to injection profiles during oilfield development is one of the most important factors that affect the oilfields’ performance. Since injection profiles are affected by multiple geological and development factors, it is difficult to model their complicated, non-linear relationships using conventional approaches. In this paper, two adaptive-network-based fuzzy inference systems (ANFIS) based neuro-fuzzy systems are presented. The two neuro-fuzzy systems are: (1) grid partition based fuzzy inference system (FIS), named ANFIS-GRID, and (2) subtractive clustering based FIS, named ANFIS-SUB. We compare the performance of resultant FIS and study the effect of parameters. A real-world injection profile data set from the Daqing Oilfield, China is used. FIS are generated and tested using training and testing data from that data set. The impact of data quality on the performance of FIS is also studied. Experiments demonstrate that although soft computing methods are somewhat of tolerant of inaccurate inputs, cleaned data results in more robust models for practical problems. ANFIS-GRID outperforms ANFIS-SUB due to its simplicity in parameter selection and its fitness in the target problem.

© 2007 Elsevier Inc. All rights reserved.

Keywords: ANFIS; Subtractive clustering; Grid partition; Petroleum industry; Data quality problems

1. Introduction

In water flooding oilfields, injected water pushes petroleum fluid (oil, gas or/and water) to move toward to wellbore through the porous media underground. Injection profiles of injection wells present the distribution of injected water in the active or producing strata. Understanding injection profiles significantly aids in analyzing production related problems, such as residual oil distribution, residual reserve estimation, water flooding efficiency, injection and production balance, and so on.
Many methods can be applied to obtain injection profiles in oilfields, such as sealed coring, sidewall coring, interpretation of logging data, C/O spectral logging, numerical simulation and comprehensive analysis of static and dynamic data from the oilfield development. Most of those methods, except for numeric simulation, are for obtaining injection profiles by in-place measurement and interpretation. They are expensive and time-consuming. In addition, it is impossible to obtain injection profiles whenever and wherever they are needed for improving oil recovery (IOR) purposes. Reservoir numeric simulation models the oil/gas production by combining petroleum fluid flow and other models. By properly modeling reservoir and matching the history production data, reservoir simulation generates injection profiles in the production history and predicts injection profiles in the future. However, reservoir simulation has its own inherent problems, including that (1) it is difficult to model multiple parameters and integrate sub-models; (2) history matching is still largely a trial-and-error, and consequently time-consuming process which depends heavily on reservoir simulation expertise; and (3) reservoir simulators sometimes encounter difficulties in modeling actual reservoir features due to their built-in limitations. In addition, time-consuming post-processing is required to obtain injection profile data from reservoir simulation results. Considering that injection profiles are required in many different IOR projects, it is desirable to have handy data available when it is required.

Soft computing techniques are known for their efficiency in dealing with complicated problems when conventional analytical methods are infeasible or too expensive, with only sets of operational data available. Soft computing methods have been widely applied in many areas in the petroleum industry, such as reservoir description [27], well logging interpretation [16], production prediction [29] and treatment optimization [17]. In this paper, two neuro-fuzzy systems, ANFIS-GRID and ANFIS-SUB, are employed to model the relationships of injection profiles and their influential parameters. A set of data from real injection profiles in the Daqing Oilfield of China is employed to train and test these neuro-fuzzy systems. Average prediction accuracy of about 80% is achieved.

The rest of this paper is organized as follows. Section 2 describes the injection profile modeling problem; Section 3 is a brief introduction to ANFIS, ANFIS-GRID, and ANFIS-SUB; Section 4 studies the effects of parameters for these neuro-fuzzy systems and presents the experimental results on the raw data; Section 5 demonstrates the effect of low quality data on the performance of FIS and presents the improved result using cleaned data; Section 6 concludes the paper.

2. Problem statement

In water flooding oilfields, the injection profile is tightly related to fluid flow in the underground porous media. Therefore, water injectivity of the injection wells is affected by many parameters. For example, the larger the permeability, the larger the water injectivity is. Water always breaks through along the high permeability channels to the producing wells. Formation communication between injection and producing wells, which depends on the depository environment for oil/gas generation and transportation, is another important factor that affects the injectivity. With a nice communication environment, stored oil/gas volume in reservoirs can be produced easily, hence nice injectivity.

In the underground porous media, petroleum fluid flow follows the non-linear Darcy’s Law, described by the following equation:

\[ u = -\frac{k}{\mu} \frac{dP}{dx}, \]

where \( u \) is the superficial velocity; \( k \) is the permeability; \( \mu \) is the viscosity of petroleum fluid; and \( dP/dx \) is the pressure drop in fluid flow direction. Considering the complex interaction of rock and fluid properties, anisotropy of permeability, the fluid flow can be generally described as follows:

\[
\begin{align*}
\nabla \cdot \left\{ \frac{k_g \kappa_g}{\mu_o \kappa_o} (\nabla p_o - \gamma_o \nabla d) \right\} + \nabla \cdot \left\{ \frac{k_w \kappa_w}{\mu_w \kappa_w} (\nabla p_w - \gamma_w \nabla d) \right\} &\pm Q_o = \phi_o \frac{\partial}{\partial t} \left( \frac{S_o}{\kappa_o} \right) \quad \text{for oil}, \\
\nabla \cdot \left\{ \frac{k_g \kappa_g}{\mu_o \kappa_o} (\nabla p_o - \gamma_o \nabla d) \right\} &\pm Q_o = \phi_o \frac{\partial}{\partial t} \left( \frac{S_o}{\kappa_o} \right) \quad \text{for water}, \\
\n\nabla \cdot \left\{ \frac{k_g \kappa_g R_o}{\mu_o \kappa_o} (\nabla p_o - \gamma_o \nabla d) \right\} + \nabla \cdot \left\{ \frac{k_w \kappa_w}{\mu_w \kappa_w} (\nabla p_w - \gamma_w \nabla d) \right\} &\pm (R_o Q_o + Q_g) = \phi_o \frac{\partial}{\partial t} \left( \frac{S_o}{\kappa_o} + \frac{R_o S_o}{\kappa_o} \right) \quad \text{for gas},
\end{align*}
\]

(2)
where $V$ is the differential functions in the spatial dimensions, and $\nabla (f) = \frac{\partial f}{\partial x} \Delta x + \frac{\partial f}{\partial y} \Delta y + \frac{\partial f}{\partial z} \Delta z$; $K$ stands for the absolute permeability of producing formations; $K_{rw}, K_{ro}, K_{rg}$ denote the relative permeability of water, oil, and gas, in multiple phase compressible flow, respectively; $\mu_o, \mu_w, \mu_g$ are viscosity of oil, water and gas underground, respectively; $B_o, B_w, B_g$ are the volume factors for oil, water, and gas, respectively; $p_o, p_w, p_g$ are the pressure of oil, water, and gas phases, respectively; $r_o, r_w, r_g$ are specific gravity of oil, water, and gas, respectively; $S_o, S_w, S_g$ are saturation of oil, water and gas, respectively; and $R_s$ is the solvable gas ratio in the oil phase.

Eq. (2) reveals the complexity of fluid flow in the porous media underground, which partly explains the complicated nature of injection profile modeling and prediction using conventional approaches such as reservoir simulation. Therefore, soft computing based modeling is proposed.

2.1. Parameter selection

As discussed above, injection profile prediction is a complicated problem that involves multiple interacting factors. In order to build a reasonably accurate model for prediction, proper parameters must be selected. Following are some practical considerations in parameter selection:

- The selected parameters must affect the target problem, i.e., strong relationships must exist among the parameters and target (or output) variables
- The selected parameters must be well-populated, and corresponding data must be as clean as possible. Since the soft computing methods model problems based on available data, the availability and quality of data are both essential

For modeling and predicting injection profiles, studies [28] have been conducted in the Daqing Oilfield, China. They select three parameters to construct the fuzzy membership functions and fuzzy rules based on analysis results of 25 wells (totally 218 active strata) and their expertise. These parameters are: (1) sand type, which is reflected by main sand, subordinate sand or untabulated stratum sand thickness; (2) connection status, which is reflected by the correspondence of sand types near the injection and producing wells; and (3) well spacing, which is the distance of an injection well and surrounding producing wells.

Formation permeability of active strata is a key factor that affects the injection profiles. Studies on available data show that absolute permeability of active strata is positively related to the sand type in the Daqing Oilfield, China [28]. Sand types can be represented by sand thickness and communication of injection and producing wells, as shown in Table 1. In addition to the positive association, permeability is not widely available in the tested area. Therefore, permeability is not considered in modeling and prediction.

Based on their research results, considering the difficulties in identifying fuzzy membership functions and domain expertise in constructing proper FIS, we select following parameters in our problem modeling:

- Gross pay thickness near the wellbore of injection wells, in meters, denoted as $h_{\text{gross}1}$
- Net pay thickness near the wellbore of injection wells, in meters, denoted as $h_{\text{net}1}$
- Gross pay thickness near the wellbore of nearby producing wells, in meters, denoted as $h_{\text{gross}2}$
- Net pay thickness near the wellbore of nearby producing wells, in meters, denoted as $h_{\text{net}2}$
- Spacing distance between injection wells and surrounding producing wells, in meters, denoted as $d$

Among these five parameters, the first four reflect the depository environment and the communication between injection and producing wells. Well spacing distance reflects the effect of well pattern and production

<table>
<thead>
<tr>
<th>Sand thickness (m)</th>
<th>&lt;0.5 gross</th>
<th>0.5–1.5 net</th>
<th>1.0–1.5 net</th>
</tr>
</thead>
<tbody>
<tr>
<td>Average permeability ($\mu$m$^2$)</td>
<td>0.037</td>
<td>0.123</td>
<td>0.264</td>
</tr>
</tbody>
</table>

Table 1
Relationship of permeability and pay zone thickness in the active producing strata [28]
criteria. The larger the well spacing, the smaller the injection capability is. Data for these parameters are available from the in-place measurement.

2.2. Problem formulation

The relationship of injection profile and the selected parameters is not obvious. In this paper, injection profile is calculated by summing up relative water injectivity of producing wells perforated in each active stratum, formulated as follows:

\[
ri_i = RI \times \text{ratio}_i \left( \sum_i\text{ratio}_i \right)_i
\]

\[
\text{ratio}_i = h_{\text{net}2_i} + (h_{\text{gross}2_i} - h_{\text{net}2_i}) / 3.3,
\]

where \( i = 1, 2, \ldots \) refers to one of the producing wells surrounding an injection well; and \( RI \) is the injectivity of a producing stratum. The water injectivity of an active stratum is calculated as follows:

\[
RI_k = \sum_i ri_i
\]

where \( k \) is the index of a producing stratum of an injection well, and \( i \) is the index of surrounding producing wells of the injection well in the producing stratum.

In order to model a system that is lack of complete or computationally feasible analytic description, soft computing methods can be used. In the case of injection profile, the parameters are \( ri_i, h_{\text{gross}1_i}, h_{\text{net}1_i}, h_{\text{gross}2_i}, h_{\text{net}2_i}, \) and \( d_i \). Available profile data is used for constructing the model. The resultant model is then validated by independent data sets. Prediction accuracy is calculated by comparing the difference of predicted and measured relative injectivity. If the difference is within tolerance, as in \( |RI_k^{\text{predicted}} - RI_k^{\text{measured}}| \leq \varepsilon \), accurate prediction is achieved. The tolerance \( \varepsilon \) is defined based on accuracy requirement in the petroleum industry. In injection profile prediction, errors of \( \pm 2\% \) are allowed. So the prediction accuracy is defined as follows:

\[
\text{accuracy} = \frac{\|RI_k^{\text{predicted}} - RI_k^{\text{measured}}\|}{\|\text{predicted set}\|} \leq \varepsilon \times 100.
\]

2.3. Sample injection profile data

Injection profiles from 10 wells are used for the problem modeling. Fig. 1 shows the complicated relationships of five selected parameters and the relative injectivity in each active stratum per producing well. It is

![Fig. 1. Relationships of selected parameters to relative injectivity of producing oil strata, where gross1 stands for \( h_{\text{gross}1} \), net1 for \( h_{\text{net}1} \), gross2 for \( h_{\text{gross}2} \), net2 for \( h_{\text{net}2} \), and well spacing for \( d \).](image-url)
obvious that non-linear relationships exist between $h_{\text{gross1}}$, $h_{\text{net1}}$, $h_{\text{gross2}}$, $h_{\text{net2}}$, $d$ and $ri$, making it difficult to build a model using conventional approaches such as regression and curve fitting.

3. Neuro-fuzzy systems

3.1. Fuzzy logic and fuzzy inference systems

Fuzzy logic (FL) and fuzzy inference systems (FIS), first proposed by Zadeh [31], provide a solution for making decisions based on vague, ambiguous, imprecise or missing data. FL represents models or knowledge using IF–THEN rules in the form of “if $X$ and $Y$ then $Z$”. As shown in Fig. 2, a fuzzy inference system mainly consists of fuzzy rules and membership functions and fuzzification and de-fuzzification operations. By applying the fuzzy inference, ordinary crisp input data produces ordinary crisp output, which is easy to be understood and interpreted. A more generalized description of fuzzy problems and uncertainty is provided in [32].

Broadly speaking, there are two categories of fuzzy inference systems, namely Mamdani [19] and Takagi–Sugeno (ST) [26] FIS. A Mamdani FIS consists of simple rules such as

IF pressure is high and temperature is low, then volume is small,

where pressure and volume and temperature are linguistic variables; high and small and low are linguistic values that are characterized by membership functions. ST type of fuzzy rules only involves fuzzy sets or membership functions in the premise part. A FIS has two inputs and two ST rules can be generally represented as follows:

$$
\begin{align*}
R^1: & \text{if } x_1 \text{ is } A_1^1 \text{ and } x_2 \text{ is } A_2^1, \text{ then } f_1 = p_1 x_1 + q_1 x_2 + c_1, \\
R^2: & \text{if } x_1 \text{ is } A_1^2 \text{ and } x_2 \text{ is } A_2^2, \text{ then } f_2 = p_2 x_1 + q_2 x_2 + c_2
\end{align*}
$$

Eq. (7) represents the first order ST type fuzzy rules. The output part can also be constants, named as Takagi–Sugeno–Kang fuzzy model [25], represented as

$$
\begin{align*}
R^1: & \text{if } x_1 \text{ is } A_1^1 \text{ and } x_2 \text{ is } A_2^1, \text{ then } f_1 = C_1, \\
R^2: & \text{if } x_1 \text{ is } A_1^2 \text{ and } x_2 \text{ is } A_2^2, \text{ then } f_2 = C_2.
\end{align*}
$$

For complicated problems as discussed in this paper, the first order ST FIS is widely employed to model the relationships of inputs and outputs.

3.2. FIS identification and refinement

Identification of the rule base is the key of a fuzzy inference system. The problems are: (1) there are no standard methods for transforming human knowledge or experience into the rule base; and (2) it is required to further tune the membership functions (MF) to minimize the output errors and to maximize the performance, as stated in [9]. There are many methods [27,20,14] that can be applied to identify the MF and FIS. In this paper, two commonly used methods are applied for FIS identification and refinement.
3.2.1. **ANFIS**

ANFIS is a multi-layer adaptive network-based fuzzy inference system proposed by Jang [9]. An ANFIS consists of totally five layers to implement different node functions to learn and tune parameters in a FIS using a hybrid learning mode. In the forward pass, with fixed premise parameters, the least squared error estimate approach is employed to update the consequent parameters and to pass the errors to the backward pass. In the backward pass, the consequent parameters are fixed and the gradient descent method is applied to update the premise parameters. Premise and consequent parameters will be identified for MF and FIS by repeating the forward and backward passes. ANIFS has been widely used in automation control [20] and other areas.

3.2.2. **ANFIS-GRID**

The ANFIS-GRID fuzzy inference system is the combination of grid partition and ANFIS. Grid partition divides the data space into rectangular sub-spaces using axis-paralleled partition based on pre-defined number of membership functions and their types in each dimension, as shown in Fig. 3. Premise fuzzy sets and parameters are calculated using the least square estimate method based on the partition and MF types. When constructing the fuzzy rules, consequent parameters in the linear output MF are set as zeros. Hence it is required to identify and refine parameters using ANFIS. The combination of grid partition and ANFIS has been reported in [1,15].

The wider application of grid partition in FL and FIS is blocked by the curse of dimensions, which means that the number of fuzzy rules increases exponentially when the number of input variables increases. For example, if there are averagely $m$ MF for every input variable and a total of $n$ input variables for the problem, the total number of fuzzy rules is $m^n$. It is obvious that the wide application of grid partition is threatened by the large number of rules. According to [10,13], grid partition is only suitable for cases with small number of input variables (e.g. less than 6). In this paper, the injection profile modeling problem has exactly five antecedent variables. It is reasonable to apply the ANFIS-GRID.

3.2.3. **ANFIS-SUB**

The ANFIS-SUB fuzzy inference system combines the subtractive clustering method and ANFIS. The subtractive clustering method is proposed by Chiu [2] by extending the mountain clustering method [30]. It clusters data points in an unsupervised way by measuring the potential of data points in the feature space. When there is not a clear idea how many clusters there should be used for a given data set, it can be used for estimating the number of clusters and the cluster centers. Subtractive clustering assumes that each data point is a potential cluster center and calculates the potential for each data point based on the density of surrounding data points. Then data point with highest potential is selected as the first cluster center, and the potential of data points near the first cluster center (within the influential radius) is destroyed. Then data points with the highest remaining potential as the next cluster center and the potential of data points near the new cluster

![Fig. 3. Grid partition of an input domain with two input variables and two membership functions for each input variable.](image-url)
center is destroyed. The influential radius is critical for determining the number of clusters. A smaller radius leads to many smaller clusters in the data space, which results in more rules, and vice versa. Hence it is important to select proper influential radius for clustering the data space.

After clustering the data space, the number of fuzzy rules and premise fuzzy MF are determined. Then the linear squares estimate is used to determine the consequent in the output MF, resulting in a valid FIS. As described above, ANFIS learns and refines the premise fuzzy MF and consequents using the least squares estimate and back propagation. Tuned by ANFIS, the resultant FIS achieves minimum training errors.

The combination of ANFIS and subtractive clustering has been widely applied in automation control [12], function approximation [3] and resolving engineering problems [4,7,11,22].

4. Injection profile modeling and predication

In injection profile modeling, we apply neuro-fuzzy inference systems, and split the sample data set based on injection wells, each training set containing data from nine wells and each testing set containing data from one well. Ten training and testing data sets are generated to check the performance of selected FIS.

When modeling practical problems using neuro-fuzzy systems, it is important to obtain proper training and testing data sets. If they are not selected properly, the testing data does not validate the model obtained using the training data, as shown in Fig. 4. For the second training and testing data, although the checking errors (the curve of chk 2) are not very large compared with the curve of chk 4, the minimum checking error (MCE) is achieved within the first epoch. This fact discovers that the checking data presented to ANFIS for training is sufficiently different from the training data set. Hence, the trained FIS does not capture the features of the testing data set very well. It is required to change the membership function types or the number of membership functions to retrain the model. Properly selected training and checking sets should have training error curves as trn 4 and chk 4, shown in Fig. 4. The checking error decreases with training proceeding until a jump point. Overfitting occurs when training passes that point. The problem is considered when constructing FIS using ANFIS, ANFIS-GRID and ANFIS-SUB.

4.1. Behavior of the ANFIS

When generate a FIS using ANFIS, it is important to select proper parameters, including the number of MF num_MF for each individual antecedent variable. It is also important to select proper parameters for the learning and refining process, including the initial step size S, the step size increase rate R_inc, and the step size decrease rate R_dec. Parameter selection and their impact on the ANFIS have been addressed in the literature [6,11,22]. For specific training and testing data sets, we analyze the effect of these parameters on the final ANFIS performance, including the training and testing MCE.

Figs. 5–8 present the impact of parameters (e.g. num_MF, S, R_Dec, and R_Inc) on the training/checking errors. From Fig. 5, combination of num_MF can affect the training and testing errors significantly; and increasing the number of MF does not necessarily improve the performance of FIS. In our case, there are
Fig. 5. Training and testing errors using different combination of radii. The initial step size $S = 0.01$, step size decrease rate $R_{Dec} = 0.9$, and step size increase rate $R_{Inc} = 1.1$. The radius combination is: $[2 2 2 2 2 2]$ for trn1 and chk1, $[2 2 2 3 2]$ for trn2 and chk2, $[2 2 3 2 2]$ for trn3 and chk3, and $[3 2 2 2 2]$ for trn4 and chk4.

Fig. 6. Training and checking errors using different initial step sizes. The constant radius $R_c = 2$, the step decreasing rate $R_{Dec} = 0.8$ and step increasing rate $R_{Inc} = 1.10$. The initial step size is: $S = 0.01$ for trn1 and chk1, $S = 0.03$ for trn2 and chk2, $S = 0.05$ for trn3 and chk3, $S = 0.07$ for trn4 and chk4.

Fig. 7. Training and checking errors using different step increasing rates. Initial step size $S = 0.01$ and step increasing rate $R_{Inc} = 0.8$. The step size decreasing rate is: $R_{Dec} = 1.05$ for trn1 and chk1, $R_{Dec} = 1.10$ for trn2 and chk2, $R_{Dec} = 1.15$ for trn3 and chk3, $R_{Dec} = 1.20$ for trn4 and chk4.
335 data points in the training set, and a FIS with 48 fuzzy rules (having totally 321 parameters) is the limitation. The best combination of radius is \([2 \, 2222]\). That means a constant radius \(R_c\) in every dimension of data space.

Fig. 6 shows the training and testing errors using different initial step size \(S\). Initial step size does not affect the value of MCE, the jump points in the checking error curves; while it does affect the training epochs when the MCE appears. The larger the initial step size, the earlier the MCE comes. Fig. 7 presents the effect of step size increase rate \(R_{Inc}\) on the training and checking errors. A similar conclusion can be drawn as the impact of the initial step size. The larger the increase rate, the faster FIS achieves the MCE. Fig. 8 shows that no significant difference exists for FIS obtained using different step size decrease rates.

From the above analysis, the 4th training and testing sets are applied to construct FIS and to cross test other testing sets. The parameters are selected as follows: \(S = 0.01\), \(R_{Inc} = 1.1\), and \(R_{Dec} = 0.9\). The cross checking results of prediction accuracy based on Eq. (6) are listed in Table 2. It is obvious that the testing accuracy can be high as 93%, which indicates that the trained FIS contains most of data patterns in the testing data. The overall average prediction accuracy is 78.7% for both training and testing sets.

### 4.2. Using the ANFIS-GRID

Due to the curse of dimension in the ANFIS-GRID, radius combinations in Fig. 5 are tried. Similar results are achieved as in Fig. 5. This indicates that after initializing the FIS with same architecture, ANFIS will identify and tune the parameters to achieve the least training error.

Table 3 lists the training and cross testing accuracy of 10 pairs of data sets. In the table, Fis1, Fis2, Fis3, Fis4, Fis5, Fis6, Fis7, Fis8, Fis9 and Fis10 are generated by the first, second, third, fourth, fifth, sixth, seventh, eighth, ninth, and tenth training sets, respectively. All training sets generate FIS with five inputs and 32 Takagi–Sugeno rules. The ANFIS training parameters are \(S = 0.01\), \(R_{Inc} = 1.1\), and \(R_{Dec} = 0.9\).

### 4.3. Using the ANFIS-SUB

In order to generate a proper FIS using ANFIS-SUB, it is critical to determine the proper influential radius for each dimension in the data space. It is also important to select proper values or combinations of following

![Fig. 8. Training and checking errors using different step increasing rates. The initial step size \(S = 0.01\) and step increasing rate \(R_{Inc} = 1.05\). The step size decreasing rate is: \(R_{Dec} = 0.8\) for \(trn\) 1 and \(chk\) 1, \(R_{Dec} = 0.85\) for \(trn\) 2 and \(chk\) 2, \(R_{Dec} = 0.90\) for \(trn\) 3 and \(chk\) 3, \(R_{Dec} = 0.95\) for \(trn\) 4 and \(chk\) 4.](image-url)
parameters: (1) influential radius $R_c$, which affects the clustering result directly; (2) quash factor $C_{\text{Quash}}$, which is used to multiple the given radii values to quash the potential of outlying points to be considered as part of that cluster, (3) accept ratio $R_{\text{Accept}}$, which sets the potential as a fraction of the potential of the first cluster center and above which a data point will be accepted as a cluster center, and (4) reject ratio $R_{\text{Reject}}$, which sets the potential as a fraction of the first cluster center and below which a data point will be rejected as a cluster center.

Extensive experiments are conducted to select proper combination of constant radius $R_c$, the quash factor $C_{\text{Quash}}$, the accept ratio $R_{\text{Accept}}$, and the reject ratio $R_{\text{Reject}}$, which affect the number of fuzzy rules significantly, as shown in Fig. 9. The numbers of fuzzy rules are obtained using different combinations of $C_{\text{Quash}}$, $R_{\text{Accept}}$ and $R_{\text{Reject}}$. It tells that the most critical parameter is $R_c$; the larger the radius constant, the fewer the fuzzy rules in the resultant FIS; larger $R_{\text{Reject}}$ will lead to fewer rules with other parameters the same; the larger quash factor will decrease the number of fuzzy rules.

When using ANFIS-SUB to generate FIS, it is required to have more than one fuzzy rule to refine the parameters. Hence parameters that lead to only one fuzzy rule in resultant FIS will be not valid. Cross validation results using 10 optimized FIS based on MCE criteria are presented in Table 4.

### 4.4. Discussion

From the above discussion, the performance of ANFIS-SUB is hard to predict. When the number of clusters for a given data set is unclear, it is difficult to specify the influential radius for each dimension, considering the impact of quash factor, accept ratio and reject ratio on clustering.

The performance of the ANFIS-GRID is mainly affected by the num_MF for each dimension, which can usually be determined by the data distribution and the size of training data set. For practical problems with

![Fig. 9. Effect of $R_c$, $C_{\text{Quash}}$, $R_{\text{Accept}}$, and $R_{\text{Reject}}$ on the number of fuzzy rules.](image)

<table>
<thead>
<tr>
<th>Train1/test1</th>
<th>Fis1</th>
<th>Fis2</th>
<th>Fis3</th>
<th>Fis4</th>
<th>Fis5</th>
<th>Fis6</th>
<th>Fis7</th>
<th>Fis8</th>
<th>Fis9</th>
<th>Fis10</th>
</tr>
</thead>
<tbody>
<tr>
<td>Train2/test2</td>
<td>82/63</td>
<td>79/87</td>
<td>81/78</td>
<td>80/78</td>
<td>80/79</td>
<td>79/93</td>
<td>80/81</td>
<td>80/84</td>
<td>82/69</td>
<td>80/84</td>
</tr>
<tr>
<td>Train3/test3</td>
<td>77/83</td>
<td><strong>80/59</strong></td>
<td>77/83</td>
<td>78/70</td>
<td>78/74</td>
<td>76/93</td>
<td>78/75</td>
<td>76/88</td>
<td>79/69</td>
<td>78/79</td>
</tr>
<tr>
<td>Train4/test4</td>
<td>76/76</td>
<td>76/74</td>
<td><strong>79/44</strong></td>
<td>75/77</td>
<td>76/79</td>
<td>73/93</td>
<td>75/81</td>
<td>75/82</td>
<td>77/67</td>
<td>76/79</td>
</tr>
<tr>
<td>Train5/test5</td>
<td>79/76</td>
<td>80/74</td>
<td>79/83</td>
<td><strong>80/56</strong></td>
<td>79/74</td>
<td>77/93</td>
<td>78/81</td>
<td>78/86</td>
<td>80/72</td>
<td>79/84</td>
</tr>
<tr>
<td>Train6/test6</td>
<td>83/80</td>
<td>84/80</td>
<td>83/81</td>
<td>84/74</td>
<td>84/74</td>
<td>82/95</td>
<td>82/85</td>
<td>82/92</td>
<td>84/74</td>
<td>83/84</td>
</tr>
<tr>
<td>Train7/test7</td>
<td>78/80</td>
<td>78/80</td>
<td>78/81</td>
<td>79/74</td>
<td>79/63</td>
<td>76/93</td>
<td>77/66</td>
<td>77/86</td>
<td>79/69</td>
<td>78/84</td>
</tr>
<tr>
<td>Train8/test8</td>
<td>75/83</td>
<td>76/74</td>
<td>75/81</td>
<td>76/78</td>
<td>76/74</td>
<td>73/96</td>
<td>79/75</td>
<td><strong>79/58</strong></td>
<td>76/69</td>
<td>76/74</td>
</tr>
<tr>
<td>Train9/test9</td>
<td>76/73</td>
<td>75/83</td>
<td>75/81</td>
<td>76/78</td>
<td>76/68</td>
<td>74/93</td>
<td>75/75</td>
<td>75/84</td>
<td><strong>80/36</strong></td>
<td>76/84</td>
</tr>
<tr>
<td>Train10/test10</td>
<td>78/76</td>
<td>78/76</td>
<td>78/81</td>
<td>79/74</td>
<td>78/74</td>
<td>76/93</td>
<td>77/87</td>
<td>77/84</td>
<td>79/72</td>
<td><strong>81/32</strong></td>
</tr>
</tbody>
</table>

Highlighted results in bold have low testing accuracy (lower than 60%). The highlighted FIS in italics achieves best performance.
few input variables (fewer than six input variables), ANFIS-GRID can be a good choice. Table 3 shows that the overall average accuracy is up to 78% for Fis7. This is a very satisfactory result in injection profile prediction. Hence the ANFIS-GRID is chosen to construct FIS in this paper.

5. Effects of noisy data

For soft computing methods, when the prediction accuracy is discussed, it is assumed that both the data used to train the models and the testing data to make predictions are free of errors [24]. But rarely a data set is clean before extraordinary effort having been put to clean the data [23]. For the problem of injection profile prediction, with measurement errors caused by reading and equipment for all selected six parameters, especially the relative injectivity, it is not unusual to have some extreme patterns which will decrease the model accuracies. In this paper, we briefly discuss the effect of data quality on FIS using selected ANFIS-GRID.

5.1. Approximate dependencies

In this work, raw data is analyzed using approximate functional dependence mining method. An approximate functional dependency, or an approximate dependency, is a functional dependency that is almost valid with some exceptional data tuples. A functional dependency studies the relationships of attributes in one or several tables, and claims that the value of an attribute is uniquely determined by the values of some other attributes. The discovery of functional dependencies in databases leads to discovery of useful knowledge and data quality problems.

More formally, a functional dependency over a relation (or a table) is expressed as $X \rightarrow A$, where $X \subseteq R$ and $A \subseteq R$. The dependency is valid in a given relation $r$ if for all pairs of records $t, u \in r$, following statements hold: if $t[B] = u[B]$ for all $B \in X$, then $t[A] = u[A]$. A functional dependency $X \rightarrow A$ is minimal if $A$ is not functionally dependent on any proper subset of $X$. The dependency $X \rightarrow A$ is trivial if $A \in X$. The task in functional dependency mining is to find all minimal non-trivial dependencies that hold in $r$.

Approximate dependencies arise in many databases when there are natural dependencies between attributes, but some records contain errors or inconsistencies. For example, the relationship between zip code and the combination of city and state in a country. Another example is the social security number (SSN) and a corresponding person residing in the USA. Theoretically, these attributes have consistent relationships, as one person associated with one SSN, and one zip code associated with one combination of city, state in a country. But if errors are somehow introduced, the relationships between these attributes will be violated, which leads to the approximate dependencies.

The TANE algorithm [8], which deals with discovering functional and approximate dependencies in large data files, is an effective algorithm in practice. The TANE algorithm partitions attributes into equivalence partitions of the set of tuples. By checking if the tuples that agree on the right-hand side agree on the left-hand side, one can determine whether a dependency holds or not. By analyzing the identified approximate dependencies, one can identify potential erroneous data in the relation.

Table 4

<table>
<thead>
<tr>
<th>Fis1</th>
<th>Fis2</th>
<th>Fis3</th>
<th>Fis4</th>
<th>Fis5</th>
<th>Fis6</th>
<th>Fis7</th>
<th>Fis8</th>
<th>Fis9</th>
<th>Fis10</th>
</tr>
</thead>
<tbody>
<tr>
<td>Train1/test1</td>
<td>70/80</td>
<td>72/90</td>
<td>70/87</td>
<td>67/77</td>
<td>70/77</td>
<td>66/83</td>
<td>68/80</td>
<td>63/83</td>
<td>68/90</td>
</tr>
<tr>
<td>Train2/test2</td>
<td>73/87</td>
<td>74/72</td>
<td>72/67</td>
<td>70/60</td>
<td>72/61</td>
<td>67/65</td>
<td>69/65</td>
<td>67/52</td>
<td>71/67</td>
</tr>
<tr>
<td>Train3/test3</td>
<td>70/75</td>
<td>74/75</td>
<td>71/69</td>
<td>68/75</td>
<td>70/75</td>
<td>67/67</td>
<td>69/69</td>
<td>65/67</td>
<td>70/72</td>
</tr>
<tr>
<td>Train4/test4</td>
<td>70/74</td>
<td>74/70</td>
<td>72/67</td>
<td>68/74</td>
<td>71/67</td>
<td>67/72</td>
<td>69/70</td>
<td>66/59</td>
<td>71/66</td>
</tr>
<tr>
<td>Train5/test5</td>
<td>70/84</td>
<td>73/84</td>
<td>71/84</td>
<td>68/74</td>
<td>70/79</td>
<td>66/79</td>
<td>68/78</td>
<td>65/79</td>
<td>70/84</td>
</tr>
<tr>
<td>Train6/test6</td>
<td>70/77</td>
<td>72/84</td>
<td>70/79</td>
<td>68/72</td>
<td>69/84</td>
<td>66/77</td>
<td>68/74</td>
<td>64/79</td>
<td>69/79</td>
</tr>
<tr>
<td>Train7/test7</td>
<td>69/79</td>
<td>72/81</td>
<td>70/77</td>
<td>67/79</td>
<td>69/77</td>
<td>66/75</td>
<td>67/79</td>
<td>62/85</td>
<td>69/79</td>
</tr>
<tr>
<td>Train8/test8</td>
<td>71/68</td>
<td>75/66</td>
<td>72/66</td>
<td>70/62</td>
<td>72/62</td>
<td>69/54</td>
<td>69/70</td>
<td>68/48</td>
<td>72/62</td>
</tr>
<tr>
<td>Train9/test9</td>
<td>71/72</td>
<td>76/59</td>
<td>72/67</td>
<td>70/56</td>
<td>71/67</td>
<td>69/53</td>
<td>72/41</td>
<td>68/46</td>
<td>72/56</td>
</tr>
<tr>
<td>Train10/test10</td>
<td>73/32</td>
<td>75/53</td>
<td>73/42</td>
<td>70/42</td>
<td>71/53</td>
<td>68/48</td>
<td>69/57</td>
<td>66/63</td>
<td>72/42</td>
</tr>
</tbody>
</table>

Highlighted results in bold have low testing accuracy (lower than 60%).
In this paper, relationships of five pivot parameters \((h_{\text{gross1}}, h_{\text{net1}}, h_{\text{gross2}}, h_{\text{net2}}, \text{and } d)\) and relative injectivity \((ri)\) are analyzed using the TANE algorithm. For equivalence partition, \(h_{\text{gross1}}, h_{\text{net1}}, h_{\text{gross2}}, h_{\text{net2}}\) are kept in their original representation. The relative injectivity is kept in the precision of 1%. The well spacing \((d)\) is processed into discrete numbers using Eq. (9). The \(d'\) keeps the unit of \(d\), which is meter.

\[
d' = \begin{cases} 
150 & 125 \leq d < 175, \\
200 & 175 \leq d < 225, \\
250 & 225 \leq d < 275, \\
300 & 275 \leq d < 325. 
\end{cases} 
\]  

\[(9)\]

5.2. Results from TANE algorithm

After data pre-processing, four approximate dependencies are discovered, as shown in Table 5. Although all these dependencies reflect the relationships among parameters, the first dependency is the most important one because it shows that selected five parameters have consistent association relationship with the water injectivity per active layer except a few data tuples, which is a very important dependency for injection profile prediction.

To identify exceptional tuples by analyzing the approximate dependencies, it is required to investigate the equivalence partitions of both left-hand and right-hand sides of an approximate dependency. It is non-trivial work that could lead to the discovery of problematic data. By analyzing the first dependency, conflicting tuples are identified as given in Table 6. From Table 6, one can see that detected tuples contain conflicting

| Table 5 | Approximate dependencies detected using the TANE algorithm |
|---|---|---|
| Index | \(h_{\text{gross1}}, h_{\text{net1}}, h_{\text{gross2}}, h_{\text{net2}}, d' \rightarrow ri\) | Rows to delete |
| 1 | \(h_{\text{gross1}}, h_{\text{net1}}\) | 25 |
| 2 | \(h_{\text{gross1}}, h_{\text{net1}}\) | 20 |
| 3 | \(h_{\text{gross1}}, h_{\text{net1}}, h_{\text{gross2}}, d' \rightarrow h_{\text{net2}}\) | 24 |
| 4 | \(h_{\text{gross1}}, h_{\text{gross2}}, h_{\text{net2}}, d', ri \rightarrow h_{\text{net1}}\) | 23 |

| Table 6 | Conflicting tuples identified by analyzing the first approximate dependency in Table 5 |
|---|---|---|
| Index | \(h_{\text{gross1}}, h_{\text{net1}}, h_{\text{gross2}}, h_{\text{net2}}, d', ri\) |
| 1 | 0.2 0 0.4 0 150 0 |
| 2 | 0.2 0 0.4 0 150 2 |
| 3 | 0.2 0 0.6 0.5 150 0 |
| 4 | 0.2 0 0.6 0.5 150 3 |
| 5 | 0.2 0 0.4 0.2 150 0 |
| 6 | 0.2 0 0.4 0.2 150 5 |
| 7 | 0.4 0.2 0.4 0 150 0 |
| 8 | 0.4 0.2 0.4 0 150 1 |
| 9 | 0.5 0 0.4 0 150 0 |
| 10 | 0.5 0 0.4 0 150 2 |
| 11 | 0.5 0.5 0.5 0.5 150 0 |
| 12 | 0.5 0.5 0.5 0.5 150 1 |
| 13 | 0.5 0 0.8 0 150 0 |
| 14 | 0.5 0 0.8 0 150 2 |
| 15 | 0.5 0.4 1.0 0.4 150 1 |
| 16 | 0.5 0.4 1.0 0.4 150 6 |
| 17 | 0.6 0.2 0.5 0.2 200 1 |
| 18 | 0.6 0.2 0.5 0.2 200 6 |
| 19 | 1.3 1.1 1.2 0.4 150 0 |
| 20 | 1.3 1.1 1.2 0.4 150 3 |

Highlighted tuples contain obvious conflicting or erroneous information.
relationships or associations among parameters, and some of them contain severe ones. For example, as the same parameters in $h_{\text{gross}1}$, $h_{\text{net}1}$, $h_{\text{gross}2}$, $h_{\text{net}2}$, $d'$ as in tuples 5 and 6, and tuples 15 and 16, the water relative injectivity per active layer for these cases bear large difference. These tuples could create trouble for injection profile modeling and prediction. Based on domain experts’ suggestion, tuples 4, 6, 16, 18, 19 are removed from the raw data sets, and more experiments are implemented using the above methods.

5.3. Results using cleaned sample data

Repeat the experiments using ANFIS-GRID using cleaned data, better results are achieved, as shown in Table 7. The corresponding first order ST FIS is represented in Figs. 10 and 11.

Fig. 11 shows the membership functions for five input variables, and Fig. 11 lists the linear consequent equations in the format of $C_1^{\text{input}_1} + C_2^{\text{input}_2} + C_3^{\text{input}_3} + C_4^{\text{input}_4} + C_5^{\text{input}_5} + C$, where $C_1$, $C_2$, $C_3$, $C_4$, $C_5$, $C$ are coefficients as shown in the Fig. 12, respectively, and $\text{input}_1$ (for gross thickness near the injection wells), $\text{input}_2$ (for net thickness near the injection wells), $\text{input}_3$ (for the gross thickness near the producing wells), $\text{input}_4$ (for the net thickness near the producing wells) and $\text{input}_5$ (for well distance between injection wells and surrounding producing wells) represent five input variables, respectively. Combining the membership functions of inputs and output, first order ST fuzzy rules can be established, in the format of Eq. (7), expressing the relationships of $h_{\text{gross}1}$, $h_{\text{net}1}$, $h_{\text{gross}2}$, $h_{\text{net}2}$, $d$ and $r_i$. A FIS can be constructed with a set of fuzzy rules, and complicated fuzzification and defuzzification operations. With the constructed FIS, crisp output, the relative injectivity, can be calculated by feeding with proper crisp inputs.

Comparing Tables 3 and 7, the effect of data quality on FIS is clearly observed. Therefore, it is highly demanded to analyze the data quality of data sets before they are applied in soft computing modeling. Fig. 12 shows the predicted relative injectivity using Fis6 mentioned in Table 7. The overall accuracy is 86.1%.

5.4. Practical results

Fuzzy mathematical approach was applied in the Daqing Oilfield of China to predict injection profile, as reported in [28]. Promising results were achieved in understanding the residual oil distribution and improving the oil recovery in the Daqing Oilfield. In their fuzzy system, sand types, connection status of injection and production wells, and distance of injection well to the production well were taken as appraisal objects or influential factors. Each appraisal factor was classified into several categories to construct fuzzy membership functions based on the domain expertise. The target variable of relative injectivity was also classified into categories, according to the percentage of relative injectivity. Three subordinate relationship plates were constructed for membership functions of sand type, connection status and well spacing with respect to the water absorbing status. They classified the relative injectivity into five categories as 0, (0,3), [3,5), [5,7), and [7,inf) or good absorbing, bad absorbing and non-absorbing. The prediction accuracy is 71% and 79% for these two classification methods, respectively.

Table 7

<table>
<thead>
<tr>
<th>Validation results from different training and testing data sets</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fis1</td>
</tr>
<tr>
<td>Train1/test1</td>
</tr>
<tr>
<td>Train2/test2</td>
</tr>
<tr>
<td>Train3/test3</td>
</tr>
<tr>
<td>Train4/test4</td>
</tr>
<tr>
<td>Train5/test5</td>
</tr>
<tr>
<td>Train6/test6</td>
</tr>
<tr>
<td>Train7/test7</td>
</tr>
<tr>
<td>Train8/test8</td>
</tr>
<tr>
<td>Train9/test9</td>
</tr>
<tr>
<td>Train10/test10</td>
</tr>
</tbody>
</table>

The training and testing sets are cleaned based on results in Table 6. Highlighted results in bold have low testing accuracy (lower than 70%). Highlighted FIS in italics has best performance.
With intensive domain expertise, it is possible to construct proper FIS as in [28]. But for relative absorbing percentage over 7%, the classification is blurred. It makes the prediction of water injection in high level difficult. With about 10% data points having water absorbing no less than 7%, it needs more detailed classification.

In addition, it is usually difficult to construct proper fuzzy membership functions, even with the assistance of domain experts. Therefore, self-learning and refinement are the best choices which enable to learn FIS from available training data sets. In order to test the resultant FIS, injection profile data from N2-D2-B447 is applied using Fis6 in Table 7. The N2-D2-B447 is an independent injection well in the South II District of the Daqing Oilfield, China. From its injection profile data, it has thin strata having large injectivity, such as $S_{24a}$ and $S_{25}$, and thick strata having small injectivity, such as $S_{216}$ and $P_{21a}$. From the results in Fig. 13, these strata are poorly predicted. The overall prediction accuracy is up to 85%.

---

**Fig. 10.** The generalized bell-shaped membership functions of five input variables using Fis6 in Table 7.
Tested by data from the Daqing Oilfield, the average injection profile prediction accuracy is improved. The new approach can be expected to have wider application in resolving complicated petroleum exploration and development related problems. Due to the easiness in FIS identification and modification, more knowledge can be saved for a real knowledge base in certain development unit or locality. Significant savings in production cost and improvement in work efficiency can be achieved.

---

Fig. 11. Linear output coefficients of fuzzy rules of fis6 in Table 7.

Fig. 12. Prediction of the sample data set using Fis6 in Table 7.

Fig. 13. Prediction injection profile for N2-D2-B447 using Fis6 mentioned in Table 7.

Tested by data from the Daqing Oilfield, the average injection profile prediction accuracy is improved. The new approach can be expected to have wider application in resolving complicated petroleum exploration and development related problems. Due to the easiness in FIS identification and modification, more knowledge can be saved for a real knowledge base in certain development unit or locality. Significant savings in production cost and improvement in work efficiency can be achieved.
6. Conclusions

As an efficient neuro-fuzzy system, ANFIS can be applied to learn FIS and to identify and refine the antecedent and consequent parameters in MF and fuzzy rules using training data sets. It provides an effective approach for many complicated engineering problems in various fields. In this paper, studies on ANFIS, ANFIS-GRID and ANFIS-SUB indicate that selection of appropriate neuro-fuzzy systems depends on the problem and available data sets. Taha and Noureldin found out that [22], in their cases, different selections of num_MF does not affect the performance of ANFIS significantly; while initial step size S and step change rates $R_{\text{Inc}}$ and $R_{\text{Dec}}$ are significant to the training RMSE of the model. In contrast, in our experiments, $R_{\text{Dec}}$ does not matter much on results.

ANFIS-GRID is known for the “curse of dimensionality”. It works best in our problem, compared with ANFIS and ANFIS-SUB. In problem modeling using ANFIS-GRID, it is important to investigate the size of the training set and the architecture of FIS, with the size of the training set being larger than the total number of parameters (e.g. for premise fuzzy sets and linear output) in the FIS.

In this paper, for five input variables, generalized bell-shaped membership functions are applied. An easier approach is applied to construct FIS based on available data sets, compared with the manual work done in [28]. The prediction accuracy is higher based on our results.

There are other types of membership functions [21] that should be tried for complicated engineering problems. Because ANFIS-GRID has its own disadvantages for problems with more than five input variables, it is also recommended to improve the performance of ANFIS-SUB by applying genetic algorithms as in [7] for parameter selection and other improvement, as seen in [5,18].

Acknowledgements

The authors thank an anonymous referee for the insightful comments, which helped to improve the paper greatly. Financial support from the New Mexico Petroleum Recovery Research Center, a research division of New Mexico Tech, is gratefully acknowledged.

References


