A NEW APPROACH FOR ESTIMATING COMPRESSION FACTOR OF NATURAL GAS BASED ON ARTIFICIAL NEURAL NETWORK

M.R. Nikholgh1, A.R. Moghadassi1, F. Parvizian1

1. Department of Chemical Engineering, Faculty of Engineering, Arak University, Phone/Fax: +(98) 861-2225946, Shariati St., Farmandari Sq., Arak - Markazi, Iran.

• E-mail: a-moghadassi@araku.ac.ir

Abstract

In this work, the ability of Artificial Neural Network or ANN based on back-propagation algorithm to modeling and predicting of compressibility factor of natural gas has been investigated. The MSE analysis based on results, are used to verifying the suggested approach. Results show, a good agreement between experimental data and ANN predictions. An important feature of the model is its needlessness to any theoretical knowledge or human experience during the training process. This work clearly shows the ability of ANN on calculating z-factor for natural gas only based on the experimental data, instead of using equations of state.

Keywords: Artificial Neural Network, Compressibility factor, natural gas.

1. Introduction

Natural gas is a subcategory of petroleum that is a naturally occurring, complex mixture of hydrocarbons, with a minor amount of inorganic compounds [1]. Knowledge of the pressure-volume-temperature (PVT) behavior of natural gases is necessary to solve many petroleum engineering problems, designing and analyzing natural gas production and processing systems. Gas reserves, gas metering, gas pressure gradients, pipeline flow and compression of gases are some of the problems requiring precise calculation of gas density.

Due to intermolecular forces, real gases do not behave ideally, particularly at elevated pressures, and ideal gas law is extended to real systems by including a compressibility factor, z. Compressibility factor, is a key parameter, which can be determined from various theoretical empirical equation of state or a generalized chart for gases.

Artificial neural network is a model based on some experimental results that is proposed to predict the required data because of avoiding more experiments. This model provides a connection between input and output variables and bypass underlying complexity inside the system. The ability to learn the behavior of the data generated by a system certifies versatility of neural network [1]. Speed, simplicity, and capacity to learn are the advantages of ANN compared to classical methods. This model has been widely applied to estimate the physical and thermodynamic properties of chemical compounds.

ANN has recently been used to predict some pure substances and petroleum fraction’s properties [2], activity coefficients of isobaric binary systems [3], dew point pressure [4], thermodynamic properties of refrigerants [5,6], and activity coefficient ratio of electrolytes in amino acid’s solutions [7], etc. However, few publications are available in literature for ANN applications in predicting PVT properties.

Defining the ANN and selecting the best ANN predictor to predict the compressibility factor in desired temperature and pressure ranges instead of empirically derived correlations are the main focus of this work. Finally results of the ANN model is evaluated against with the unseen data and then compared with the empirical models.

2. Compressibility Factor Theory

All gases behave ideally when the pressure approaches zero. Due to intermolecular forces, real gases do not behave ideally, particularly at elevated pressures. Eq. (1) is extended to real systems by including a compressibility factor, z, as:

\[ P_v = zRT \] (1)

Gas compressibility factor is also called deviation factor, or z-factor. In fact, its value reflects how much the real gas deviates from the ideal gas at given pressure and temperature. Definition of the compressibility factor is expressed as:
The gas compressibility factor can be determined on the basis of measurements in PVT laboratories. For a given amount of gas, if temperature is kept constant and volume is measured at 14.7 psia and an elevated pressure $P_1$, $z$-factor can then be determined with the following formula:

$$z = \frac{V_1}{14.7 V_0}$$

where $V_0$ and $V_1$ are gas volumes measured at 14.7 psia and $P_1$, respectively.

Very often the $z$-factor is estimated with the chart developed by Standing and Katz [8]. This chart has been set up for computer solution by a number of individuals. Brill and Beggs yield $z$-factor values accurate enough for many engineering calculations. Hall and Yarborough presented more accurate correlation to estimate compressibility factor of natural gas.

Nonetheless, there is no single equation that accurately estimates compressibility factor of natural gas under all conditions. Therefore the new attempts have been made to develop an alternative to a simple method which can be used for all conditions.

3. ARTIFICIAL NEURAL NETWORKS

In order to find relationship between the input and output data derived from experimental works, a more powerful method than the traditional ones are necessary. ANN is an especially efficient algorithm to approximate any function with finite number of discontinuities by learning the relationships between input and output vectors [9]. These algorithms can learn from the experiments, and also are fault tolerant in the sense that they are able to handle noisy and incomplete data. The ANNs are able to deal with non-linear problems, and once trained can perform estimation and generalization rapidly [10]. They have been used to solve complex problems that are difficult to be solved if not impossible by the conventional approaches, such as control, optimization, pattern recognition, classification, and so on, specially it is desired to have the minimum difference between the predicted and observed (actual) outputs [11].

Artificial neural networks are biological inspirations based on the various brain functionality characteristics. They are composed of many simple elements called neurons that are interconnected by links and act like axons to determine an empirical relationship between the inputs and outputs of a given system. Multiple layers arrangement of a typical interconnected neural network is shown in Figure (1). It consists of an input layer, an output layer, and one hidden layer with different roles. Each connecting line has an associated weight.

![Neural Network Architecture](image)

Artificial neural networks are trained by adjusting these input weights (connection weights), so that the calculated outputs may be approximated by the desired values. The output from a given neuron is calculated by applying a transfer function to a weighted summation of its input to give an output, which can serve as input to other neurons, as follows [12].

$$\alpha_k = F_k \left( \sum_{i=1}^{N_{hid}} w_{ik} \alpha_{i(k-1)} + \beta_{ik} \right)$$

$$z = \frac{V_{actual}}{V_{ideal \ gas}}$$

$$z = \frac{P_1 V_1}{14.7 V_0}$$
Where $\alpha_{jk}$ are $j$th neuron outputs from $k$th layer and $\beta_{jk}$ is the bias weight for neuron $j$ in layer $k$. The model fitting parameters $w_{jk}$ are the connection weights. The nonlinear activation transfer functions $F_k$ may have many different forms. The classical ones are threshold, sigmoid, Gaussian and linear function, etc. [13]. For more details of various activation functions see [14].

The training process requires a proper set of data i.e. input ($I_i$) and target output ($t_i$). During training the weights and biases of the network are iteratively adjusted to minimize the network performance function [15]. The typical performance function that is used for training feed forward neural networks is the network Mean Squares Errors (MSE) Eq. (5).

$$MSE = \frac{1}{N} \sum_{i=1}^{N} (e_i)^2 = \frac{1}{N} \sum_{i=1}^{N} (t_i - \alpha_i)^2$$

There are many different types of neural networks, differing by their network topology and/or learning algorithm. In this paper the back propagation learning algorithm, which is one of the most commonly used algorithms is designed to predict the compressibility factor of natural gas. Back propagation is a multilayer feed-forward network with hidden layers between the input and output [16]. The simplest implementation of back propagation learning is the network weights and biases updates in the direction of the negative gradient that the performance function decreases most rapidly. An iteration of this algorithm can be written as follows [17]:

$$x_{k+1} = x_k - \alpha_k g_k$$

where $x_k$ is a vector of current weights and biases, $g_k$ is the current gradient, and $\alpha_k$ is the learning rate.

There are various back propagation algorithms such as Scaled Conjugate Gradient (SCG), Levenberg-Marquardt (LM) and Resilient back Propagation (RP). LM is the fastest training algorithm for networks of moderate size and it has the memory reduction feature to be used when the training set is large. One of the most important general purpose back propagation training algorithms is SCG.

The neural nets learn to recognize the patterns of the data sets during the training process. Neural nets teach themselves the patterns of the data set letting the analyst to perform more interesting flexible work in a changing environment. Although neural network may take some time to learn a sudden drastic change, but it is excellent to adapt constantly changing information [17]. However the programmed systems are constrained by the designed situation and they are not valid otherwise. Neural networks build informative models whereas the more conventional models fail to do so. Because of handling very complex interactions, the neural networks can easily model data, which are too difficult to model traditionally (inferential statistics or programming logic). Performance of neural networks is at least as good as classical statistical modeling, and even better in most cases [18]. The neural networks built models are more reflective of the data structure and are significantly faster.

Neural networks now operate well with modest computer hardware. Although neural networks are computationally intensive, the routines have been optimized to the point that they can now run in reasonable time on personal computers. They do not require supercomputers as they did in the early days of neural network research.

4. RESULTS AND DISCUSSION

Developing the neural network model to accurately predict the $z$-factor of natural gas requires its exposure to a large data set during the training phase. A set of data containing pseudo-reduced pressure, pseudo-reduced temperature and compressibility factor was collected from Handbook of Natural Gas Engineering [8]. Table (1) lists samples of these data which were used for training and testing the neural network.

<table>
<thead>
<tr>
<th>Properties</th>
<th>min</th>
<th>max</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pseudo-Reduced Pressure</td>
<td>0.2</td>
<td>15</td>
</tr>
<tr>
<td>Pseudo-Reduced Temperature</td>
<td>1.05</td>
<td>3</td>
</tr>
<tr>
<td>Compressibility Factor</td>
<td>0.351</td>
<td>1.707</td>
</tr>
</tbody>
</table>

The back propagation method with LM, SCG and RP learning algorithm has been used in feed forward, single hidden layer network. Input layer neurons have no transfer functions. The compressibility factor depends only on $T_{pr}$. 

---

Table 1: Minimum and Maximum of Data which using in ANN
Consequently, inputs are the pseudo-reduced temperature and pseudo-reduced pressure while output is the compressibility factor.

Neural network training can be made more efficient if certain preprocessing steps are performed on the network inputs and targets [19]. All of the inputs (before feeding to the network) and output data in training phase, were used for pre-processing by normalizing the inputs and targets in the range [-1, 1].

The neurons in the hidden layer perform two tasks: summing the weighted inputs connected to them and passing the result through a nonlinear activation function to the output or adjacent neurons of the corresponding hidden layer. Training and testing of the ANN was performed with the toolbox of MATLAB.

More than 80% of data set is used to train each ANN and the rest have been used to evaluate their accuracy and trend stability. Number of hidden neurons has been systematically varied to obtain a good estimate of the trained data. The selection criteria is the net output MSE. The MSE of various hidden layer neurons are shown in Figure (2). As it can be seen the optimum number of hidden layer neurons is determined to be 100 for minimum MSE. Similarly the MSE of various training algorithms was calculated and listed in Table (2) for the obtained 100 hidden layer neurons. As Table (2) shows the Levenberg-Marquardt (LM) algorithm has the minimum MSE.

<table>
<thead>
<tr>
<th>Training Algorithm</th>
<th>MSE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Trainlm</td>
<td>0.000635</td>
</tr>
<tr>
<td>Trainscg</td>
<td>0.047500</td>
</tr>
<tr>
<td>Trainrp</td>
<td>0.163400</td>
</tr>
</tbody>
</table>

According to these tables the Levenberg-Marquardt (LM) algorithm is the most suitable algorithm with the minimum MSE. Consequently, LM provides the best minimum error average for both training and testing of the network. Figure (3) shows the LM algorithm relative error fluctuations.

The results show that the ANN can predict compressibility factor very close to the experimentally measured ones. Figures (4) and (5) show the scatter diagrams that compare the experimental data versus the computed neural network data for both training and testing the ANN.
As it may be seen, a tight cloud of points about the 45° line is obtained for the new data points. This indicates an excellent agreement between the experimental and the calculated data. Figure (6) compares the ANN simulation and experimental data.

![Comparison between predicted data by ANN and Experimental data](image)

**Fig. 6: Comparison between ANN and Exp. Data for Compressibility Factor**

5. CONCLUSION

Nowadays, there is no single equation that accurately estimates compressibility factor of natural gas under all conditions [20]. In this work, the ability of Artificial Neural Network based on back-propagation algorithm to modeling and predicting of compressibility factor of natural gas has been investigated. Learning algorithm and number of hidden layer neurons have been systematically varied to obtain a good estimate of the trained data. More than 80% of data set is used to train each ANN and the rest have been used to evaluate their accuracy and trend stability. The MSE analysis based on results, are used to verifying the suggested approach.

The optimum number of hidden layer neurons is determined to be 100 for minimum MSE. Also, the Levenberg-Marquardt algorithm is the most suitable algorithm with the minimum MSE. Consequently, LM provides the best minimum error average for both training and testing of the network. These results have been shown in previous figures. Results show, a good agreement between experimental data and ANN predictions.

REFERENCES