Iranian Journal of Chemical Engineering
Vol. 3, No. 4 (Autumn), 2006, IAChE

A hybrid neural–genetic algorithm for predicting pure and impure CO_2 minimum miscibility pressure

*S. A. Mousavi Dehghani[*1], M. Vafaie Sefti[2], A. Ameri[2], N. Shojai Kaveh[3]*

1- Research Institute of Petroleum Industry, NIOC, RIPI, Tehran, Iran.
2- Chemical Engineering Department, Tarbiat Modares University, Tehran, I.R. Iran.
3- Chemical Engineering Department, Iran University of Science and Technology, Tehran, I.R. Iran.

Abstract

Accurate prediction of the minimum miscibility pressure (MMP) in a gas injection process is crucial to optimizing the management of gas injection projects. Because the experimental determination of MMP is very expensive and time-consuming, searching for a fast and robust mathematical determination of CO_2-oil MMP is usually requested. This paper presents a new model based on a hybrid neural-genetic algorithm for predicting pure and impure CO_2-oil MMP. The CO_2-oil MMP of a reservoir fluid was correlated with the reservoir temperature, the composition of the oil, and that of the solution gas. The developed model is able to reflect the impacts on the CO_2–oil MMP of the molecular weight of the C_{5+} fraction, reservoir temperature, and solution gas in the oil. The validity of this new model was successfully approved by comparing the model results to the calculated results for the common pure and impure CO_2-oil MMP correlations. The new model yielded the accurate prediction of the experimental slim-tube CO_2-oil MMP with the lowest mean absolute percentage error (MAPE), the standard deviation of error (SD), the root mean square error (RMSE), and the highest correlation coefficient among tested impure and pure CO_2-oil MMP correlations. The results demonstrate that the hybrid neural-genetic model can be applied successfully and provide high accuracy and reliability for MMP forecasting.

Keywords: *Forecasting, Minimum miscibility pressure, Neural-genetic model, Gas injection*

Introduction

Gas injection above the minimum miscibility pressure (MMP) is a widely practiced means for improving oil recovery in many reservoirs. The minimum miscibility pressure is the lowest pressure for which a gas can develop miscibility through a multicontact process with a given reservoir oil at reservoir temperature. Reservoir pressures below the MMP result in immiscible displacements and, consequently, lower oil recoveries. At or above the MMP, miscibility can develop through a vaporizing process, a condensing process, or sometimes a combination of the two processes [1]. In the vaporizing gas process, intermediate molecular weight

* - Corresponding author: mousavisa@ripi.ir

hydrocarbons from crude oil are transferred to the leading edge of the gas front, enabling it to become miscible with the reservoir crude. In the condensing gas process, the injected gas is enriched with light hydrocarbons, usually LPGs. The reservoir oil left behind the gas front is enriched by the net transfer of the light hydrocarbons from the gas phase into the oil. Enrichment of the reservoir oil proceeds until it becomes miscible with the injected rich gas. Miscibility can also develop through a combination of the vaporizing and the condensing processes. With this combination of condensation and vaporization, miscibility may never completely develop, yet the process can result in low residual oil saturations [2].

From an experimental point of view, the MMP is routinely determined by slim tube displacements. In those experiments a long (say 10 m or longer), small diameter (say 0.5 cm) tube packed with sand or glass beads is filled with oil that is then displaced by injection gas at a fixed temperature and pressure. Typically, recovery increases rapidly with increasing pressure and then levels off. The MMP is usually taken to be the intersection of the lines drawn through the recovery points in the steeply climbing and level regions as long as the recovery in the level region is above some arbitrary cutoff (often 90%). To accurately determine the MMP, it is generally necessary to perform six displacements at six different pressures. The time required to perform a displacement is about 8 h (a working day). This means one week of experimental work is necessary to measure one MMP. The cost of such a work depends on the salary of the engineer who makes the experiment, but may be evaluated at 10,000 US dollars [3]. It is thus clear that, for petroleum companies, the slim tube test is a very expensive experiment.

Among gas injection processes, CO_2 is preferred to hydrocarbon gases because of its lower cost and high displacement efficiency [4]. To facilitate screening procedures and to gain insight into the miscible displacement process, many correlations relating the MMP to the physical properties of the oil and displacing gas have been proposed [5-11]. However, the main concern with statistical techniques such as multiple linear and nonlinear regression techniques is the difficulties in satisfying many rigid assumptions that are essential for justifying their applications, such as those of sample size, linearity, and continuity [12]. Therefore, nonlinear modeling techniques such as artificial neural networks are necessary for building an accurate and reliable predictive model. Huang et al., [13] developed an artificial neural network for predicting MMP. The MMP was correlated with the reservoir temperature, molecular weight of the C_{5+} fraction, volatile oil fraction (CH_4+N_2) to intermediate oil fraction (C_2 to C_4, H_2S, and CO_2), and composition of the CO_2 stream. Also, the impure CO_2 MMP factor was predicted by correlating the concentration of contaminants (N_2, C_1, H_2S and SO_2) in the CO_2 stream and their critical temperatures.

When artificial neural networks are used for prediction and forecasting, the underlying philosophy is similar to that used in traditional statistical approaches. Therefore, ANNs and statistical models are closely related. Consequently, the principles that are considered good practice in the development of statistical models need to be given careful consideration. The major areas that should be addressed include data pre-processing, choice of adequate model inputs, choice of an appropriate network geometry, parameter estimation, and model validation. At each stage, a number of alternatives are available to modelers. This offers great flexibility, but can also create difficulties as there are no clear guidelines to indicate under what circumstances particular approaches should be adopted. Therefore, the performance is very much dependent on the network architecture. Hence, an optimum or near optimum network structure is of utmost importance. This can be done by means of

genetic algorithms (GA) as a powerful optimization tool.

The design of neural networks using GA principles can be very helpful in terms of two main issues [14]:

- It automates the design of the network which will otherwise have to be done by hand using trial and error.
- The process of design can be analogous to a biological process in which the neural network blueprints encoded in chromosomes develop through an evolutionary process.

In this paper, the advantage of a neural-genetic computing technique in modeling the prediction of MMP in a gas injection process is examined. The prediction begins by generating ANNs that define the relationship between the input/output data. The optimization of ANNs set is then done with genetic algorithms as a powerful optimization tool. Finally, the performance of the neural-genetic model is compared to the conventional methods by means of some statistical indices.

Theoretical background

Neural networks

Neural networks are powerful tools for the approximation of unknown nonlinear functions and have gained wide applications in a variety of fields [15]. A typical three layer feed-forward neural network (FNN) is briefly depicted in Fig. 1. The neural network is able to learn the underlying relationship from a collection of training samples. The most famous training algorithm is error back propagation algorithm (BP). Consider a neural network with M layers, where the first layer is named input layer, the last layer named output layer, and the others named hidden layers. The lth layer has n_l neurons (numbered from #1 to #n_l) and a threshold neuron (numbered #0). Given S training samples ($\boldsymbol{U}_s,\boldsymbol{T}_s$), s = 1,2,. . . ,S, where $\boldsymbol{U}_s=[u_1^s,...u_{n_1}^s]$ and $\boldsymbol{T}_s=[t_1^s,...t_{n_M}^s]$. For a certain input $\boldsymbol{U}=[u_1,...u_{n_1}]$ of the network, the input x_i^l and output y_i^l of ith neuron of lth layer are as follows:

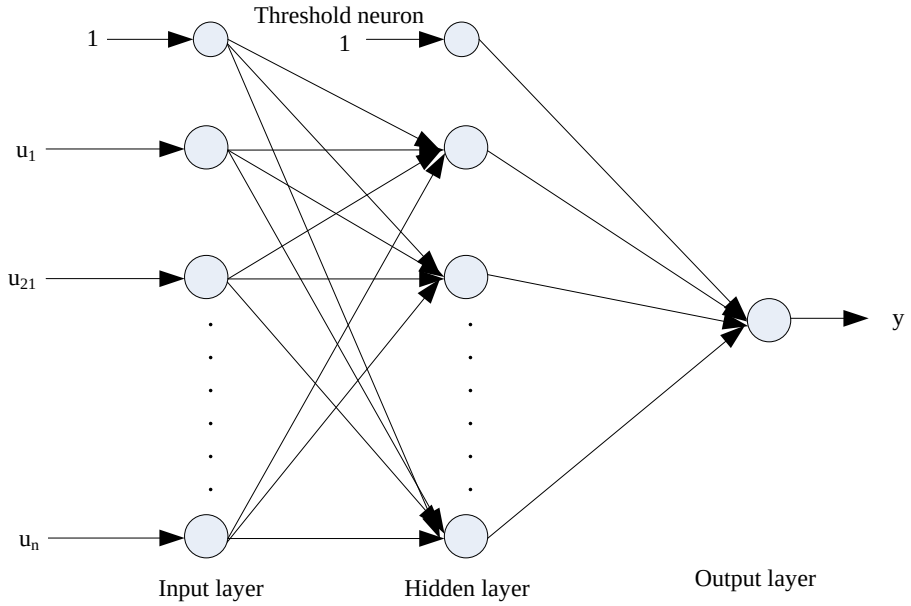

Figure 1. Brief structure of multi-layer feed-forward ANN [16].

Input layer:

$$\begin{cases} y_0^1 = x_0^1 \equiv 1 \\ y_j^1 = x_j^1 = u_j \qquad j = 1,2,...,n_1 \end{cases} \tag{1}$$

Hidden layer:

$$\begin{cases} y_0^l = x_0^l \equiv 1 \qquad j = 1,2,...,n_l ; l = 1,2,...,M-1 \\ x_j^l = \sum_{i=0}^{n_{l-1}} w_{i,j}^{l-1,l} y_i^{l-1} \\ y_j^l = g(x_j^l) = 1/[1+exp(-x_j^l)] \end{cases} \tag{2}$$

where $w_{i,j}^{l-1,l}$ is the weight from the *i*th neuron of the (*l*-1)th layer to the *j*th neuron of the *l*th layer, and $g(\cdot)$ is active function.

Output layer:

$$\begin{cases} x_j^M = \sum_{i=0}^{n_{M-1}} w_{i,j}^{M-1,M} y_i^{M-1} \\ y_j^M = g(x_j^M) = 1/[1+exp(-x_j^M)] \end{cases} \tag{3}$$

Training a neural network is to minimize the error function by determining the weights $w_{i,j}^{l-1,l}$. Once the ANN has been trained, it can be used to predict the unknown output of some input. The widely used training error function is as follows:

$$E = \sum_{s=1}^{S} E_s = \frac{1}{2} \sum_{s=1}^{S} \sum_{i=1}^{N_M} (t_i^s - y_i^{s,M})^2$$

where $y_i^{s,M}$ is the output of the *i*th neuron of the *M*th layer when inputting the *s*th sample. The classic BP updates the weight by following the summarized rule.

$$w_{i,j}^{l-1,l}(k+1) = w_{i,j}^{l-1,l}(k) - \lambda . \partial E / \partial w_{i,j}^{l-1,l} \tag{4}$$

where $\lambda \in (0,1)$ is the learning rate.

$$\partial E / \partial w_{i,j}^{l-1,l} = \sum_{s=1}^{S} \delta_j^{s,l}(k) . y_i^{s,l-1}(k) \tag{5}$$

where

$$\begin{cases} \delta_j^{s,l}(k) = [y_j^{s,l} - t_j^s] . g'[x_j^{s,l}(k)], \quad l = M \\ g'[x_j^{s,l}(k)] = \sum_{m=1}^{N_{l+1}} \delta_m^{s,l+1}(k) . w_{j,m}^{l,l+1}(k) \quad l = M\text{-}1,...,2 \end{cases} \tag{6}$$

and $g'(\cdot)$ is the derivative of active function.

To further improve the performance of the BP algorithm, the following BPM algorithm (BP with momentum) [16] is often used.

$$w_{i,j}^{l-1,l}(k+1) = w_{i,j}^{l-1,l}(k) - \lambda . \partial E / \partial w_{i,j}^{l-1,l}(k) + \beta [w_{i,j}^{l-1,l}(k) - w_{i,j}^{l-1,l}(k-1)] \tag{7}$$

where β is the momentum factor.

Genetic algorithms

Based on the idea of ‘‘survival of the fittest’’ and ‘‘natural selection’’, GA is a class of parallel iterative algorithm with a certain learning ability that repeats evaluation, selection, crossover and mutation after initialization until the stopping condition is satisfied [17]. GA is naturally parallel and exhibits implicit parallelism, which does not evaluate and improve a single solution, but analyses and modifies a set of solutions simultaneously. Even if the selection operator can select some ‘‘good’’ solutions as seeds with random initialization, the crossover operator can generate new solutions, hope-fully retaining good features from parents, and the mutation operator can enhance diversity and provide a chance to escape from the local optima. Due to the efficient and robust performance, GA have been deeply studied and successfully applied in many fields [4,7,18]. However, for those problems without explicitly unknown forms

of objective functions, the approximate fitness value should be provided efficiently for the GA to guide the evolutionary search.

Genetic evolve neural networks

The back propagation neural network, which is one of the main algorithms for learning neural networks in petroleum engineering applications, does have some drawbacks. Some of these drawbacks are the slowness of learning speed, possibility of falling into local minimum and the necessity of adjusting a learning constant in every application. To address some of these drawbacks, the genetic based design of neural networks has been proposed [19]. A major issue in the genetic based design of a neural network is that of representation (encoding). The encoding should be capable of capturing all of the important aspects of the problem. Therefore in GA, the representation scheme should be capable of allowing new, meaningful and valid network architecture to be produced by the genetic operators, (like crossover or mutation). GA are applied to the neural network in two different ways:

- They either employ a fixed network structure with a connection under evolutionary control.
- They are used in designing the structure of the network.

Therefore the evolution that has been introduced to neural networks can be divided roughly into different levels: (a) connection weights; (b) architecture; (c) learning rules.

In the application of GA to the training of neural networks, the parameters of the problem are encoded as a set of chromosomes called the population, and candidate solutions are assigned fitness values based on the constraints of the problem. Based on each individual's fitness, a selection mechanism selects a mate with a high fitness value for genetic manipulation. The manipulation process uses standard crossover and mutation genetic operators to produce a new Population of individuals (offspring) [17]:

- The *selection* is based on fitness, that is, the fitter an individual, the greater the chance of getting selected for reproduction.
- *Crossover* operator takes two chromosomes and swaps part of their genetic information to produce a new chromo-some.
- *Mutation* is implemented by occasionally altering a random bit in a string before the offspring are inserted into the new population.

Fig. 2 shows the working principles of GA. In the neural network application of GA, one can distinguish between direct encoding and indirect encoding. When all the details can be specified by chromosomes, this is called a direct encoding; if other details cannot, this is called indirect encoding. The indirect encoding method is biologically more plausible than the direct encoding method. This is because it is impossible for genetic information encoded in chromosomes to specify the whole nervous system. Fig. 3 shows an example of encoding information of neural networks for GA application. Fig. 3(*a*) shows an example neural network to be trained genetically, and Fig. 3(*b*) is chromosome representation of a neural network in Fig. 3(*a*). A neural network consists of 12 weights. Therefore 12 weights can be encoded and the length of the string is based on the number of bits per weight. The number of bits to be used to represent each weight can have a significant effect. If too few bits are used the effect of weight quantization will be significant, resulting in poor convergence. On the other hand, a large number of bits per weight will again lead to slow convergence because of a long chromosome string.

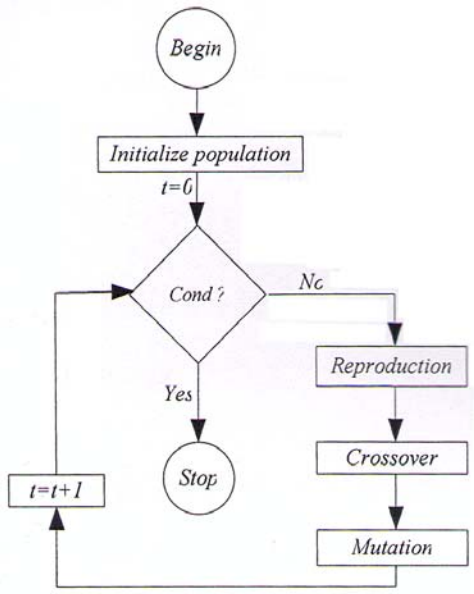

Figure 2. A flowchart of working principle of genetic algorithm [17].

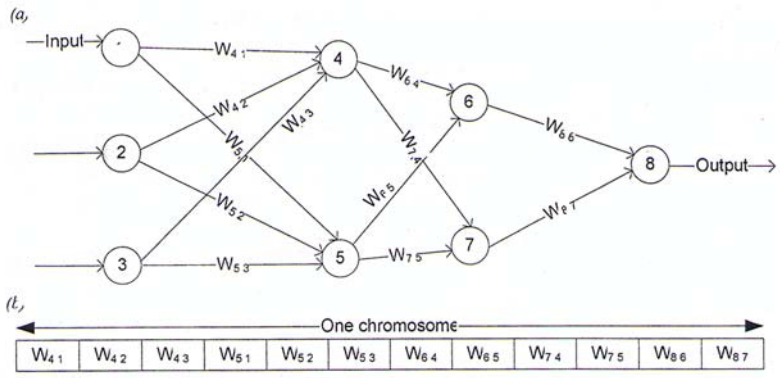

Figure 3. (*a*) Example of an artificial neural network; (*b*) genetic representation of (*a*).

Developing pure and impure CO_2-Oil MMP Model

For CO_2 MMP modeling, reservoir temperature T_R, molecular weight of C_{5+} fraction, ratio of volatile (X_{vol}) to intermediate (X_{int}) oil fraction, and the pseudocritical temperature of the injected gas are selected as input variables. In the case of pure CO_2, pseudocritical temperature will be pure CO_2 critical temperature. The data used for developing the neural-genetic model are from Jacobson [20], Graue and Zana [21], Gardner et al. [22], Frimodig et al. [23], Cardenas et al. [24], Alston et al. [9], Metcalfe [25] and, Dong et al. [6]. The experimental data reported in the literature that was used to develop and validate the models are presented in Table 1. Out of the total 55 input/output data sets, 44 data pairs were used for training the model. The model was trained for 50 epochs. To validate the model, 11 data sets were used for the testing purpose. Fig. 4 shows the proposed flowchart of a neural-genetic based algorithm for pure

and impure CO_2 MMP modeling. Back propagation neural network was assumed for all the runs. The searching mechanism of the implemented hybrid strategy can be briefly described as follows. Firstly, ANNs are constructed with a BP training algorithm based on a collection of training samples. In order to optimize the network structure, the parameters of the network, such as the number of neurons in the hidden layers, momentum and learning coefficients are treated as variables. Here, the number of neurons in the two hidden layers are coded as binary variables and are allowed to take only integer values. The other two parameters, learning and momentum coefficients, are coded as real variables and are allowed to take real values. The fitness is evaluated after training the network for specific iterations, which is kept as the mean square error (MSE) of the network based on the comparison between measured and predicted MMP values. A Roulette Wheel Parent Selection is carried out to select two parents (chromo-somes) from the population to produce two children (new chromosomes) by the reproduction operators. This method is selected because it is faster than other methods (e.g., tournament parent selection). A one-point crossover with a probability (P(c)) equal to 100% has been used. After crossover and production of two children chromosomes, one gene is selected from each child chromosome to mutate its value (mutation probability (P(m)) is equal to 1%) by adding a random value to its old value as follows:

$$New\ value = \beta \times Old\ value + \gamma \times Randomvalue$$

$$0 \le \beta \le 1\ and\ 0 \le \gamma \le 1$$

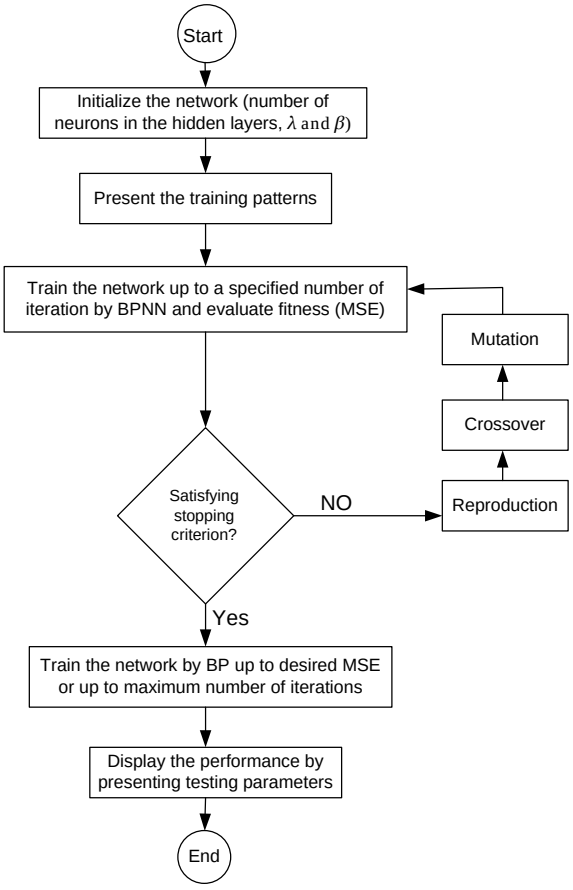

Figure 4. Flowchart of the hybrid neural-genetic algorithm.

Table 1. Experimental CO_2-oil MMP data from different literature sources.

	Composition of CO_2 stream					T_R	Oil composition		Experimental MMP
Ref.	CO_2 %	C_1 %	N_2 %	C_2-C_4 %	H_2S %	$^\circ C$	MW_{C5+}	*Vol./Int.*	**(Mpas)**
9	100	0	0	0	0	54.44	185.83	0.6	10.34
9	100	0	0	0	0	61.11	185.83	0.14	10.34
9	100	0	0	0	0	57.78	202.61	0.42	11.71
12	90.5	0	0	9.5	0	71.11	221	5.9	18.61
12	100	0	0	0	0	54.44	185.83	0.14	9.48
12	100	0	0	0	0	71.11	221	5.9	23.43
12	100	0	0	0	0	37.78	235.56	2.58	16.54
12	100	0	0	0	0	112.22	213.5	1.16	24.13
12	100	0	0	0	0	54.44	185.83	0.67	10.34
12	92.5	7.5	0	0	0	54.44	185.83	0.14	10.34
12	100	0	0	0	0	110	180.6	0.91	20.19
12	90	10	0	0	0	54.44	185.83	0.73	13.09
12	95	0	0	5	0	71.11	221	5.9	18.61
12	86.4	10.7	0	2.9	0	73.33	227	7.71	23.09
22	87.5	0	6.3	6.2	0	76.11	227	7.71	23.15
22	79.2	0	8.8	12	0	76.11	227	7.71	23.15
22	100	0	0	0	0	67.78	203.81	1.35	16.88
23	100	0	0	0	0	42.78	196.1	0.82	10.61
24	90	10	0	0	0	54.44	171.2	0.93	12.4
24	100	0	0	0	0	54.44	171.2	0.93	10.98
25	80	0	0	20	0	65.56	187.27	1.51	10.49
25	45	10	0	0	45	40.56	187.8	0.74	8.82
26	50	0	0	0	50	57.22	187.8	0.74	8.96
26	90	0	0	10	0	65.56	187.27	1.51	11.03
26	100	0	0	0	0	40.56	187.8	0.74	8.27
27	80	0	0	20	0	65.56	187.27	1.51	12.87
27	60	20	0	0	20	57.22	187.8	0.74	17.23
28	100	0	0	0	0	65.56	187.27	1.51	13.44
28	90	10	0	0	0	40.56	187.8	0.74	11.07
28	80	20	0	0	0	40.56	187.8	0.74	14.86
28	100	0	0	0	0	48.89	187.27	1.51	11.03
28	80	0	0	20	0	48.89	187.27	1.51	9.65
28	50	0	0	0	50	40.56	187.8	0.74	6.53
28	80	0	0	20	0	48.89	187.27	1.51	7.92
28	80	20	0	0	0	57.22	187.8	0.74	18.61
28	67.5	10	0	0	22.5	57.22	187.8	0.74	12.4
28	60	20	0	0	20	40.56	187.8	0.74	14.06
28	90	0	0	10	0	48.89	187.27	1.51	7.89
28	90	10	0	0	0	57.22	187.8	0.74	15.33
28	90	0	0	10	0	65.56	187.27	1.51	8.96
28	40	20	0	0	40	57.22	187.8	0.74	12.4
28	45	10	0	0	45	57.22	187.8	0.74	10.37
28	90	0	0	10	0	48.89	187.27	1.51	9.3
28	67.5	10	0	0	22.5	40.56	187.8	0.74	10.25
28	100	0	0	0	0	57.22	187.8	0.74	11.71
28	100	0	0	0	0	32.22	187.8	0.74	6.89
28	75	0	0	0	25	57.22	187.8	0.74	10.3
28	90	0	0	10	0	65.56	187.27	1.51	13.02
28	90	0	0	10	0	48.89	187.27	1.51	9.99
29	40	20	0	0	40	40.56	187.8	0.74	12.09
29	75	0	0	0	25	40.56	187.8	0.74	7.53
29	92.25	0	0	7.75	0	112.22	213.5	1.16	19.67
30	100	0	0	0	0	71.11	207.9	0.32	15.51
31	95	4.9	0.1	0	0	71.11	207.9	0.32	16.81
31	100	0	0	0	0	42.78	204.1	0.82	10.34

Note.
Vol.: Volatile oil fraction; *Int.*: Intermediate oil fraction

where $\beta = 0.95$ and $\gamma = 0.35$ is selected in this work. These values are selected based on GA performance, where β is increased, and γ is decreased to detect any improvement in the fitness value. Otherwise, the process is reversed. Furthermore, this technique enables the use of a part of the last reached solution. After producing the two children chromo-somes, they are evaluated prior to being used to obtain their fitness values. The best two chromosomes (i.e., the most fit two chromosomes) from the two parents and the two children are then inserted back into the population to improve the group of solutions. Until the stopping criterion of the GA is satisfied, the strategy will output the best solution resulting from the GA and its performance determined by detailed evalua-tion based on the comparison between measured and predicted MMP values. The optimization is carried out with a population of 30 with the stopping criteria as the maximum number of generations kept at 50. The final network is then trained by BP up to the desired level of MSE and the performance is checked by presenting the unseen testing patterns.

Model performance

The performances of the models developed in this study were assessed using various standard statistical performance evaluation criteria. The statistical measures considered were Pearson's correlation coefficient (R), standard deviation (SD), root mean square error (RMSE), and mean absolute percentage error (MAPE) (Table 2).

Table **2.** List of the performance measures.

Statistical parameter	Expression
Pearson's correlation coefficient (R)	$\left[\frac{\sum_{i=1}^{n}(MMP_i^M - \overline{MMP}^M)(MMP_i^P - \overline{MMP}^P)}{\sqrt{\sum_{i=1}^{n}(MMP_i^M - \overline{MMP}^M)}\sqrt{\sum_{i=1}^{n}(MMP_i^P - \overline{MMP}^P)}}\right]$
Standard deviation (SD)	$\pm\sqrt{\frac{\sum_{i=1}^{n}\left[\left\lvert\frac{MMP_i^M - MMP_i^P}{MMP_i^M}\right\rvert \times 100\right]^2}{n-1}}$
Root mean square error (RMSE)	$\sqrt{\frac{\sum_{i=1}^{n}(MMP_i^M - MMP_i^P)^2}{n}}$
Mean absolute percentage error (MAPE)	$\frac{1}{n}\sum_{i=1}^{n}\left\lvert\frac{MMP_i^M - MMP_i^P}{MMP_i^M}\right\rvert \times 100$

Note.

MMP^M : *Measured MMP;* $\overline{MMP}^M$: *Average of measured MMP data*

MMP^P : *Predicted MMP,* $\overline{MMP}^P$: *Average of predicted MMP data*

Results and discussion

The MMP data were divided into two data sets consisting of training and validation test data. In the training phase, a larger part of the data (80%) was used to train the network and the remaining data (20%) were used in the validation phase. In order to get performance variation information, a total of five runs are performed. The variation information is shown in Table 3. As the mean of the objective function value is near the value obtained from Run 1, the final parameters of the networks structure are fixed according to Run 1. That is, a network with these specifications could produce the optimum structure for MMP forecasting. Thus, this form of network was selected. It must be mentioned that this optimum structure is in the sense of our work, thus it is possible for a different set of data to have a different optimum structure. It depends on the training data. The final value of the number of neurons in the first hidden layer is found to be 7, and that of the second hidden layer as 5. The learning and momentum coefficients are found to be 0.653 and 0.522, respectively, after optimizing by GA. The scatter plots in Figs. 5, 6 and 7 provide comparisons of the measured CO_2 MMP levels with the neural-genetic derived ones, as well as those provided by Alson et al. [9] and Emera and Sarma [4] using statistical and GA-based methods, respectively. As shown, the neural-genetic model produces lower error levels compared with the two other models. Table 4 shows the outputs of statistical analysis for calibration results from the neural-genetic, statistical and GA-based models for MMP forecasting. It is indicated that the developed neural-genetic model has lower calibration errors than that developed by Alston et al. [9] and Emera and Sarma [4]. In detail, in the training phase, the neural-genetic model improved the Alston et al. [9] model forecast of about 43% and 32% reduction in RMSE and MAPE values, respectively. In addition, improvements of the forecast results regarding the correlation coefficient (R) and standard deviation (SD) values during the training phase were approximately 3.15% and 13%, respectively. On the other hand, the neural-genetic model improved the Emera and Sarma [4] model forecast of about 59.7% and 59.13% reduction in RMSE and MAPE values, respectively. Also, for the Emera and Sarma [4] model, improvements of the forecast results regarding the correlation coefficient (R) and standard deviation (SD) values during the training phase were approximately 8.8% and 51.31%, respectively.

Table 3. Performance variation information of GA for different runs.

Run	Objective function value (MSE)	Number of neurons in hidden layer-1	Number of neurons in hidden layer-2	Learning coefficient	Momentum coefficient
1	0.348	7	5	0.653	0.522
2	0.356	7	4	0.646	0.582
3	0.360	6	5	0.685	0.622
4	0.341	5	3	0.645	0.516
5	0.337	6	4	0.654	0.565
Mean	0.348				

Table 4. Statistical analysis for calibration results from the neural-genetic and statistical models.

Method	MAPE	RMSE	SD(%)	R
Neural-genetic (This work)	3.58	0.59	5.75	0.98
Emera and Sarma [4] (GA)	5.26	1.05	6.61	0.95
Alston et al. [9] (statistical)	8.76	1.49	11.81	0.90

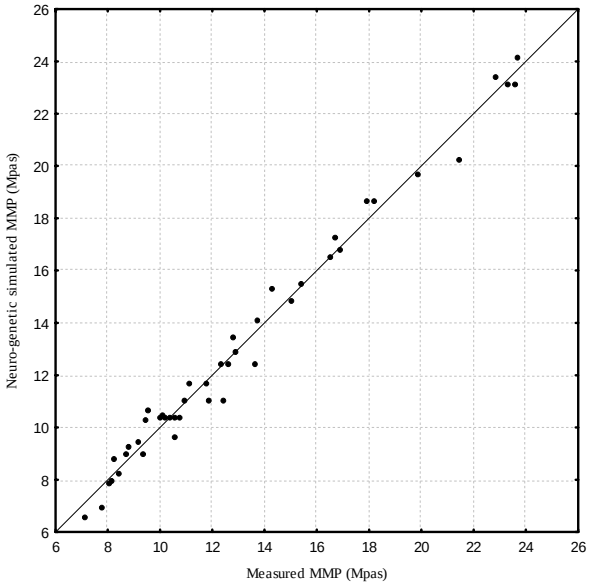

Figure 5. The measured versus neural-genetic simulated MMP values.

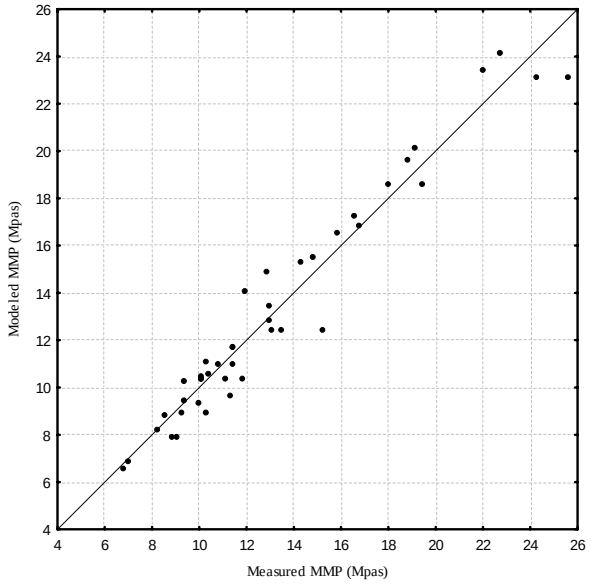

Figure 6. The measured versus modeled MMP values (from Alston et al. [9]).

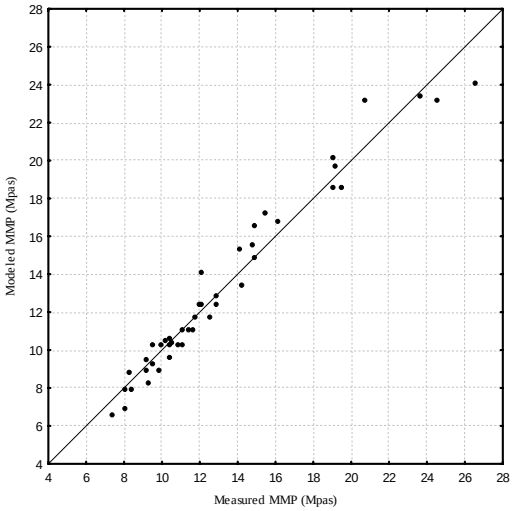

Figure 7. The measured versus modeled MMP values (from Emera and Sarma [4]).

In addition, in the validation phase as seen in Figs. 8, 9 and 10, the values with the neural-genetic model prediction were able to produce a good forecast, as compared to statistical models. Table 5 shows the results of error analysis for prediction outputs from a developed neural-genetic model and the two other models. It is indicated that outputs from the neural-genetic model are more accurate than those from the models of Alston et al. [9] and Emera and Sarma [4]. In the validation phase, the neural-genetic model improved the Alston et al. [9] model forecast of about 69.3% and 58.8% reduction in RMSE and MAPE values, respectively. In addition, improvements of the forecast results regarding the correlation coefficient (R) and standard deviation (SD) values during the validation phase were approximately 7.7% and 58.52%, respectively. On the other hand, the neural-genetic model improved the Emera and Sarma [4] model forecast of about 82% and 76.8% reduction in RMSE and MAPE values, respectively. Also, for the Emera and Sarma [4] model, improvements of the forecast results regarding the correlation coefficient (R) and standard deviation (SD) values during the training phase were approximately 16.7% and 78.6%, respectively. The major advantage of the neural-genetic approach was the ability to capture well the minor trend in the MMP series, while the two other models failed in producing good results. Thus the results indicate that the neural-genetic model is able to identify the events for which it was designed, although the extent to which this model can generalize its ability to forecast events was not included in the training process.

Table 5. Error analysis for prediction outputs.

Method	MAPE	RMSE	SD (%)	R
Neural-genetic (this work)	3.98	0.54	5.04	0.98
Emera and Sarma [4] (GA)	7.66	1.46	10.15	0.93
Alston et al. [9] (statistical)	17.12	3.12	23.57	0.84

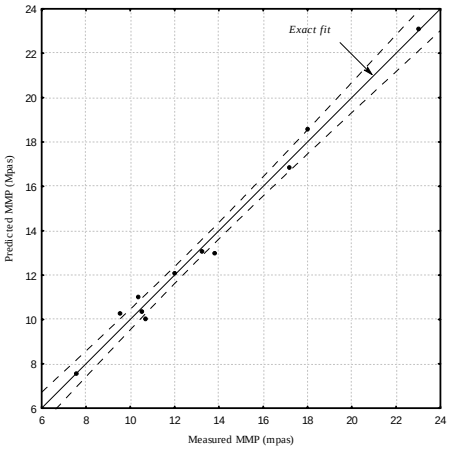

Figure 8. The measured versus neural-genetic predicted MMP values within 95% accuracy.

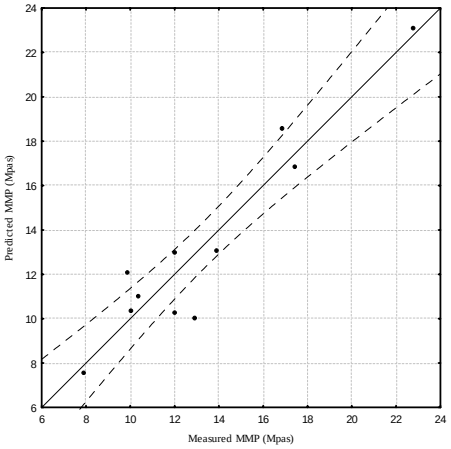

Figure 9. The measured versus statistically predicted MMP values within 95% accuracy (from Alston et al. [9]).

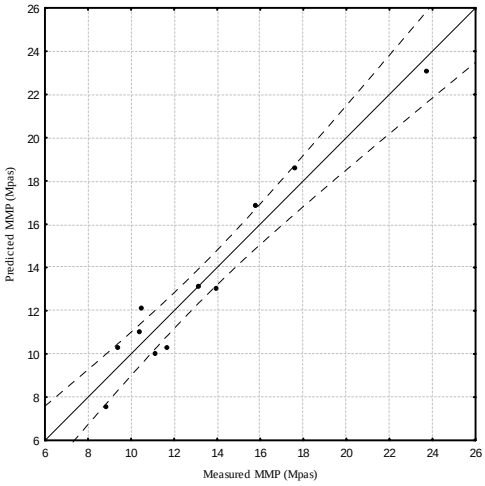

Figure 10. The measured versus statistically predicted MMP values within 95% accuracy (from Emera and Sarma [4]).

Sensitivity analysis

The analysis of variance (ANOVA) approach was used to demonstrate the sensitivity analysis of the new model and the dependence of the dependent variable (CO_2-oil MMP) on each of the independent variables. The results of the sensitivity analysis (shown in Fig. 11) are based on the rank correlation coefficient calculated between the output variable (CO_2-oil MMP) and the samples for each of the input distributions. The higher the correlation between any input variable and output variable means higher significant influence of that input in determining the output's value. From Fig. 11, it is obvious that the reservoir temperature has a major impact on the CO_2-oil MMP, and as the temperature increases, the CO_2-oil MMP increases, which confirms all published correlations. Also, the effects of oil compositions on the predicted MMP confirms all published correlations, whereas increasing MW_{C5+} or volatiles mole percent leads to an increase in the CO_2-oil MMP. On the other hand, any increase in the mole percent of the intermediate components (C_2-C_4, H_2S, and CO_2) causes a decrease in the CO_2-oil MMP. In addition, the existence of non-CO_2 components such as H_2S and hydrocarbon components (C_2 to C_4) critical temperatures greater than the CO_2 in the CO_2 stream has a positive impact on the MMP, whereas they contribute to a decrease in the MMP.

Conclusions

Miscible gas flooding is widely employed for improving or enhancing oil recovery for many oil reservoirs. A key parameter used for assessing the applicability of the process for a reservoir is the minimum miscibility pres-sure. An inaccurate prediction may result in significant consequences. For example, recommendation for a too high operating level of MMP may result in greatly inflated operation costs as well as occupational health concerns. On the other hand, if the suggested MMP is too low, the miscible displacement process would become ineffective, leading to a high risk of system failure. Thus, a higher prediction accuracy would bring significant economic benefits.

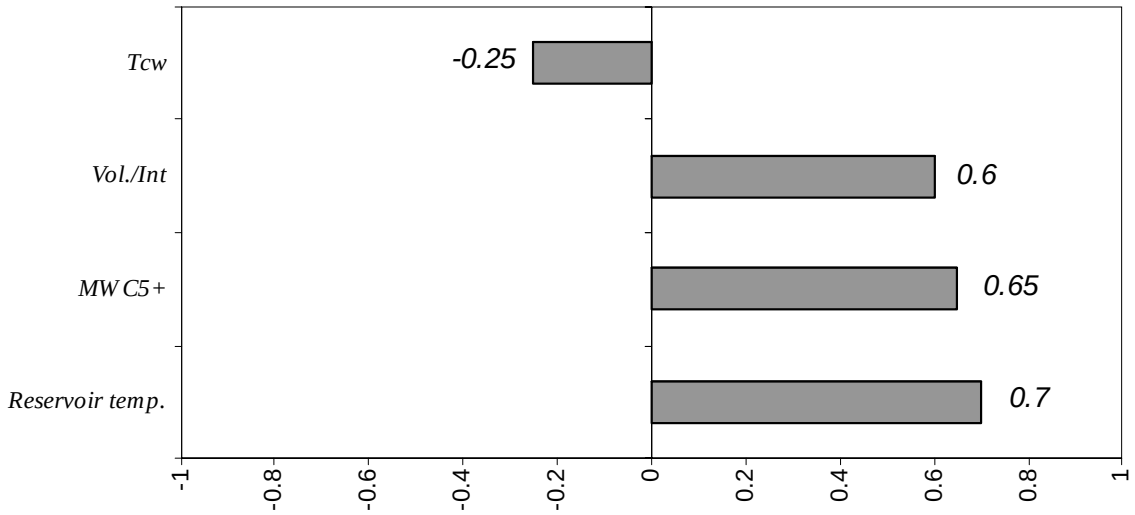

Figure 11. Relative variable impacts on CO_2 MMP.

An attempt was made in this study to investigate the application of a neural-genetic concept for the prediction of MMP in a gas injection process. In this study, the neural-genetic approach is, for the first time, used to predict MMP. The MMP data derived from the literature were employed to train and test some models. A model was successfully applied to both pure and impure CO_2 streams. In the case of the impure CO_2 stream, weight-average pseudocritical temperature was used as an indicator to show the level of contaminates that exist in the CO_2 stream. A comparison of the prediction accuracies of the neural-genetic and other statistical methods indicated that the neural-genetic approach was more accurate in predicting MMP. Thus, the results of this study suggest that the neural-genetic model is more reliable than other conventional methods for predicting MMP. The neural-genetic approach was able to produce a higher accuracy than other forcasting methods, especially under conditions with limited field information.

References

1. Zick, A. A., "A combined condensing/ vaporizing mechanism in the displacement of oil by enriched gases", SPE 15493, 1– 11 (1986).
2. Elsharkawy, A.M., "Measuring CO_2 minimum miscibility pressures: Slim-tube or Rising bubble method?", Energy & Fuels, 10, 443-449, (1996).
3. Jaubert, N.N., Avaullee, L., Pierre, C., "Is it still necessary to measure the minimum miscibility pressure?", Ind. Eng. Chem. Res., 41, 303-310, (2002).
4. Emera, M.K., Sarma, H.K., "Use of genetic algorithm to estimate CO_2–oil minimum miscibility pressure—a key parameter in design of CO_2 miscible flood", J. Pet. Sci. Eng. 46, 37– 52 (2004).
5. Shokir, Eissa M. El-M., CO_2-oil minimum miscibility pressure model for impure and pure CO_2 streams, Journal of Petroleum Science and Engineering (2007).
6. Dong, M., Huang, S., Srivastava, R., "Effect of solution gas in oil on CO_2 minimum miscibility pressure", J. Can. Pet. Technol. 39 (11), 53– 61, (2000).
7. Emera, M.K., Sarma, H.K., "Use of genetic algorithm to predict minimum miscibility pressure between flue gases and oil in design of flue gas injection project", Paper SPE 93478 Presented at the 14th SPE Middle East Oil & gas show and conference, Bahrain, 12-15March, (2005).
8. Sebastian, H.M., Wenger, R.S., Renner, T.A., "Correlation of minimum miscibility pressure for impure CO_2 streams", Paper SPE 12648 Presented at the 1984 SPE/DOE Enhanced Oil Recovery Symposium, Tulsa, April 15–18, JPT , 37 (2), 268– 274 (1985).
9. Alston, R.B., Kokolis, G.P., James, C.F., "CO_2 minimum miscibility pressures: a correlation for impure CO_2 streams and live oil systems", SPE J., 268–274, (April, 1985).
10. Yuan, H., Johns, R.T., Egwuenu, A.M., Dindoruk, B., Improved MMP correlations for CO_2 floods using analytical gas flooding theory. SPE Paper 89359 presented at the SPE/DOE Fourteenth Symposium on Improved Oil Recovery, Tulsa, USA, pp. 1-16, (2004).
11. Zuo, Y.X., Chu, J.Z., Ke, S.L., Guo, T.M., "A study of the minimum miscibility pressure for miscible flooding systems", J. Pet. Sci. Eng. 8, 315– 328, (1993).
12. Gharbi, R.B., Elsharkawy, A.M., Neural network model for estimating the PVT properties of Middle East crude oils. SPE Reserv. Evalu. Eng. 2 (3), 255–265, June, (1999).
13. Huang, Y.F., Huang, G.H., Dong, M.Z., Feng, G.M., "Development of an artificial neural network model for predicting minimum miscibility pressure in CO_2 flooding", J. Pet. Sci. Eng. 37, 83– 95, (2003).
14. Bunke, H., Kandel A., Hybrid methods in pattern recognition, World Scientific, (2002).
15. Rajasekaran, S., Vijayalakshmi Pai, G. A., Neural networks, fuzzy logic, and genetic algorithms: synthesis and applications, Prentice-Hall, (2004).
16. Harvey, R.L., "Neural network principles", Prentice Hall, (1994).
17. Vose, M.D., "The simple genetic algorithm: foundations and theory", MIT press, (1999).

18. Oswaldo, V.L., "Genetic algorithms in oil industry: An overview", Journal of petroleum science and engineering, 47, 15-22, (2005).
19. Kitano, H. "Neurogenetic learning: an integrated method of designing and training neural networks using genetic algorithms", Phys. D: Nonlinear Phenomena 75 (1–3), 225–238, (1994).
20. Jacobson, H.A., "Acid gases and their contribution to miscibility", J. Can. Pet. Technol., 56– 59 (April– June, 1972).
21. Graue, D.J., Zana, E.T., "Study of a possible CO_2 flood in Rangeley field", J. Pet. Technol., 1312–1318 (July, 1981).
22. Gardner, J.W., Orr, F.M., Patel, P.D., "The effect of phase behavior on CO_2 flood displacement efficiency", J. Pet. Technol., 2067– 2081 (Nov., 1981).
23. Frimodig, J.P., Reese, N.A., Williams, C.A., "Carbon dioxide flooding evaluation of high-pour-point, paraffinic red wash reservoir oil", Soc. Pet. Eng. J., 587–594 (Aug., 1983).
24. Cardenas, R.L., et al., "Laboratory design of a gravity-stable, miscible CO_2 process", J. Pet. Technol., 111 –118 (Jan., 1984).
25. Metcalfe, R.S., Effects of impurities on minimum miscibility pressures and minimum enrichment levels for CO_2 and rich-gas displacements, Soc. Pet. Eng. J. 4, 219–225, (1982).