DOI 10.1007/s12182-012-0200-2

Genetic algorithm application for matching ordinary black oil PVT data

Mohammad Taghizadeh Sarvestani[1], Behnam Sedaee Sola[2]* and Fariborz Rashidi[3]

[1] Department of Petroleum Engineering, Amirkabir University of Technology, Tehran, Iran
[2] Institute of Petroleum Engineering, University of Tehran, Tehran, Iran
[3] Department of Chemical Engineering, Amirkabir University of Technology, Tehran, Iran

Abstract: In the study of reservoirs, it is vital that we have a realistic physical model of the reservoir fluid that accurately describes the hydrocarbon system and its properties. The available equations of state (EOS) to model the fluid phase behavior have some inherent deficiencies that may cause erroneous predictions for real reservoir fluids, so these models should be tuned against experimental data by adjusting some parameters. Since there are many matching parameters, tuning the EOS against experimental data is a tedious and difficult work. In this study, a genetic algorithm as an optimization technique is used to solve this regression problem. This study presents a new method that uses a specially designed genetic algorithm to search for suitable regression parameters to match the EOS against measured data. The proposed method has been tested on three real black oil samples. The results show the surprising performance of the developed genetic algorithm to match the experimental data of the selected fluid samples. The main advantage of the used method is its high speed in finding a solution. Also, finding more than one solution, working automatically, confining the role of experts to the last stage, reducing costs and having the possibility of evaluating the different situations are the other advantages of this method to match ordinary black oil PVT data and makes it an ideal method to implement as an automatic EOS tuning algorithm for black oils.

Key words: Equations of state (EOS), tuning, genetic algorithms, black oil, chromosome, regression

1 Introduction

In the study of reservoirs, an accurate description of the hydrocarbon system and its properties is important. It is vital that we have a realistic physical model of our reservoir fluid before we try to use it in a reservoir simulation. Oil and gas properties are normally obtained through laboratory tests upon oil and gas samples. Experimental data are obtained under some specific conditions and usually are not sufficient for a reservoir study. On the basis of these experimental data, a model of the hydrocarbon mixture can be developed and used in combination with an equation of state model to calculate additional oil and gas properties under necessary conditions.

The available equations of state (EOS) to model the fluid phase behavior have some inherent deficiencies, particularly for multi-component mixtures, that usually cause the models to predict erroneous results even for well characterized model fluids. Real reservoir fluids, composed of thousands of compounds, are described by a limited number of pure substances and carbon groups. The carbon groups are not fully defined. Generalized correlations, often with significantly diverging results amongst themselves, are used to estimate the critical properties of the carbon groups required for EOS calculations. All these factors further deteriorate predictions of EOS for real reservoir fluids (Danesh, 1998).

The current approach in the oil industry to overcome these deficiencies is to calibrate, or tune the EOS models against experimental data. There are no well defined rules for how to do regression of an equation of state model to match to laboratory measurements. The paper by Coats and Smart (1986) contains an appendix on the choice, selection and range limits of regression variables. However, the Coats and Smart model is limited in choice of regression variables and equation of state.

Although there is no single standard method for tuning, the various approaches are basically similar; any differences between the measured and calculated data are minimized using a regression facility which adjusts various equation of state parameters. This tuned model is then can regarded as a representative of the reservoir fluid. Therefore, the main purpose is to minimize an objective function, defined as the sum of weighted squared deviations, as shown below:

*Corresponding author. email: Sedaee@ut.ac.ir
Received June 20, 2011

$$\Delta = \sum_{j=1}^{N_{\text{data}}} \left[W_j \left(\frac{\Psi_j^{\text{pred}}(X_i) - \Psi_j^{\text{exp}}}{\Psi_j^{\text{exp}}} \right) \right]^2 \quad (1)$$

where Ψ^{pred} and Ψ^{exp} represent predicted and experimental values respectively; W is the weighting factor, N_{data} expresses the number of measured data points to be fitted; and X_i designates the regression variables. The optimum values of variables are obtained by minimizing the multi-variable regression function Δ.

Since there are many matching parameters to tune the EOS against experimental data, the only practical way to perform regression is by trial and error. Therefore, it is efficient to use optimization techniques to solve the problem. Different classes of search techniques like calculus-based techniques, guided random search techniques, and enumerative techniques are developed to deal with the optimization. However, the strong non-linearity of the EOS tuning process makes classical deterministic optimization methods inefficient and unlikely to be successful. Therefore, an alternative approach would be to use heuristic type methods like genetic algorithms. Strong features of genetic algorithm such as its ability to use continuous and discontinuous variables, working in a parallel mode (Forrest et al, 1999), its ability to change several variables simultaneously and work with different data structures at the same time, suggest this optimization method to be a better choice to solve the problems of matching PVT data automatically.

Genetic algorithms (GAs) are stochastic techniques whose search methods model a natural evolution. Their approach is based on a stochastic-directed trend with roots on ideas from natural evolution and fundamental ideas of Darwin. Over the last 30 years, many engineering problems are solved by employing genetic algorithms; examples include: the traveling salesman problem (Gretenstette et al, 1985), nuclear reactor management (Poon and Parks, 1992), process control (Fogarty et al, 1995), and aircraft design (Parnee and Watson, 1999). GAs were first used in petroleum engineering in 1980 for gas pipeline operation (Goldberg, 1983). Nikravesh et al (2003) have provided a collection of articles about the application of soft computing techniques in oil exploration. Also, Velez-Langs (2005) presented many papers in the literature that describe genetic algorithm applications in seismology, well log analysis, reservoir flow simulation, hydraulic fracturing design and reservoir permeability. Romero and Carter (2001) used a GA approach to implement reservoir characterization by conditioning the reservoir simulation model to the production data on a structural model.

In this study, application of a GA to the problem of matching PVT data for three real black oil fluid samples is described. The reservoir fluids are sampled from Iranian oil reservoirs.

2 Description of the method

In this section, the method proposed in this paper to tune an EOS against the experimental data will be described. In the next section, we describe the performance of GA in selecting the regression parameters in detail.

In the commercial PVT software, an experienced user spends a lot of time to specify many different parameters in an interactive way. A suitable EOS that can be tuned against the experimental data, matching parameters of the selected EOS, binary interaction coefficients (BIC), weighting factors for different properties of the fluid which are measured experimentally, the method and number of pseudo-components if splitting of the plus fraction is needed, and a suitable model to calculate the oil viscosity and its coefficients, should be determined by the PVT expert to find the model describing the phase behavior of the reservoir fluid. Then the software uses the selected parameters to perform a multi-variable regression and by adjusting the selected matching variables, tries to find the best model for that run. At the end of each run, the PVT expert evaluates the results and decides to accept or reject the fluid model and then selects another set of items for the next run. These trial and error procedures continue until a physically acceptable match is found.

It is clear that tuning of EOS in this way is a tedious and time consuming task. In this study, almost all the processes are done automatically in a program. The program selects the EOS model, assigns weighting factors, and determines the limitations of matching parameters in a smart way and then uses the GA to find the proper viscosity model, binary interaction coefficients and matching parameters of the selected EOS and viscosity model. Also, the program checks the different cases for the number of pseudo-components if splitting of heavy components is occurring. To simplify the problem in our program, two different EOS, namely 3-parameters Peng-Robinson (3-PR) EOS and 3-parameters Soave-Redlich-Kwong (3-SRK) EOS were tested, and the program was run for each case thoroughly. Also for each selected EOS, five different values of weighting factors and three different situations regarding the number of pseudo-components were tested. These were: no splitting, 2 pseudo-components and 3 pseudo-components. In assigning the weighting factors, the general rules were applied. For example, the highest weighting factor was assigned to the saturation pressure. Also, the method of constant mole fraction (CMF) in splitting the heavy component was used. As a result, by running the program, thirty different situations will be tested. For each case, the suitable regression variables, parameters of the selected EOS, BICs, viscosity model and its parameters are specified by GA. Then, all the selected items are fed into the commercial PVT software, which has been coupled with the program, as the input data. The commercial software performs a multi-variable regression and changes the values of selected parameters in their predetermined limit ranges in an attempt to minimize the objective function and presents the best solution. The regression method used in the PVT software uses the Newton numerical method to find the minimum of a residual function that is defined as the difference between the observed data matrix and the calculated data matrix. The residual function depends on the values of calculated data that are functions of regression parameters. The output of the software is then evaluated

by the GA to modify the regression variables for the next run. This GA process is iterated for that special case until a mathematically accepted solution is obtained according to its root mean square (RMS) value. The solution saved by the program and the new fluid model replaces the original EOS and the GA process iterated for this new EOS. This cycle is repeated three times and at the end of these cycles, the program compares these 3 solutions and selects the best one with the least RMS for the special case. The function of GA will be described in the next section in details. Note that the RMS value does not always decrease through the three iterations since the weighting factors do not change through these iterations in the program. Also, modifying the matching parameters indiscriminately may results in the models that are mathematically and physically unacceptable, so it is possible that the first model is the best and modifying the model degrades it. Fig. 1 shows three different cases extracted from the program. In case a, the RMS value decreases through the iterations slowly, while in case b, the RMS value increases through the iterations. Case c shows that the RMS value decreases at the first time, but further modifying the model deteriorates it completely. As it is shown, the variations in RMS value are very sharp in case b for the third iteration relative to the second one since the model may not represent the fluid behavior any more by further modifying the

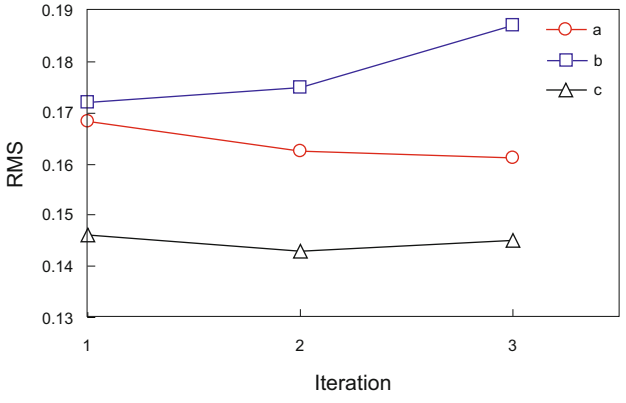

Fig. 1 RMS variations through three iterations

parameters in this case.

This process is iterated for the 30 different cases and at the end of each case, the best solution (lowest RMS) will be saved. All these processes are performed automatically in the program. At the end, 30 mathematically accepted solutions are obtained by the program.

The least-squares fit to the observation data is not necessarily the goal of equation of state fitting. To be precise, the goal is to create a fluid model that behaves like the reservoir fluid. There are aspects of the reservoir fluid that the model must capture accurately.

The phase diagram of the final model should represent the real fluid behavior. Sometimes modifying the matching parameters indiscriminately results in badly tuned models that are unrealistic physical models of our reservoir fluid and the phase diagram could not be created. Also, it is possible that the program could not converge to a favorite RMS value in a predetermined number of iterations for some special cases (e.g. the selected EOS could not model the fluid behavior or splitting the pseudo-component causes the model to diverge); therefore, it selects the best solution that may not be tuned against experimental data. Furthermore, sometimes the tuning of the model is physically unacceptable and it cannot predict the real fluid behavior at other conditions correctly, so the model cannot be used in a reservoir simulation.

Therefore, in the last stage, a PVT expert should evaluate these mathematically accepted solutions and by considering the mentioned aspects and using engineering judgment, selects the physically accepted ones. So, by use of this program, we confine the role of an expert to the last step.

The complete process is shown in Fig. 2.

3 Genetic algorithms

Genetic algorithms are heuristic type methods that were first proposed by Holland (1975) as an abstraction of biological evolution drawing on ideas from natural evolution and genetic for the design and implementation of robust adaptive systems, and since then a number of advanced forms of simple GAs have been reported by several authors (Konak et al, 2006; Bunnag and Sun, 2005). Over the last 30 years, GAs have received much attention because of their potential as optimization techniques for complex functions and an extensive number of applications (Goldberg, 1989; Michalewicz, 1992). GAs are randomized search algorithms, simulating the process of natural evolution and follows the same principles as those in nature (survival of the fittest, Charles Darwin). Although initially they were proposed as an academic investigation, today GAs have been shown to be effective over a wide range of problems. GA is applicable to multi-objective optimization and can handle conflicts among objectives. Therefore, it is robust where multiple solutions exist.

The genetic algorithms start with an initial population of randomly chosen feasible solutions to the problem being addressed from the search space. As the algorithm progresses, the current population of the solution is known as the parent population and by applying genetic operators that model the genetic processes occurring in the nature, i.e. selection, crossover and mutation, the fitter individuals are selected and their genetic information is recombined and modified to generate the new population known as offspring. The offspring are inserted into the population, replacing the parent population and producing a new generation. The genetic algorithm covers almost all parts of the solution space in each generation of search, and by evolution of the algorithm, the response point is directed to the optimum value. This process is repeated until the optimization criterion is met.

According to the described structure, the formulation of a GA for a specific problem requires the definition of three main issues: the initialization of possible solutions and representing them in a genetic format, the selection of individuals according to their fitness value, and the genetic operators such as crossover and mutation used to generate new solutions.

In the following section, these elements will be discussed

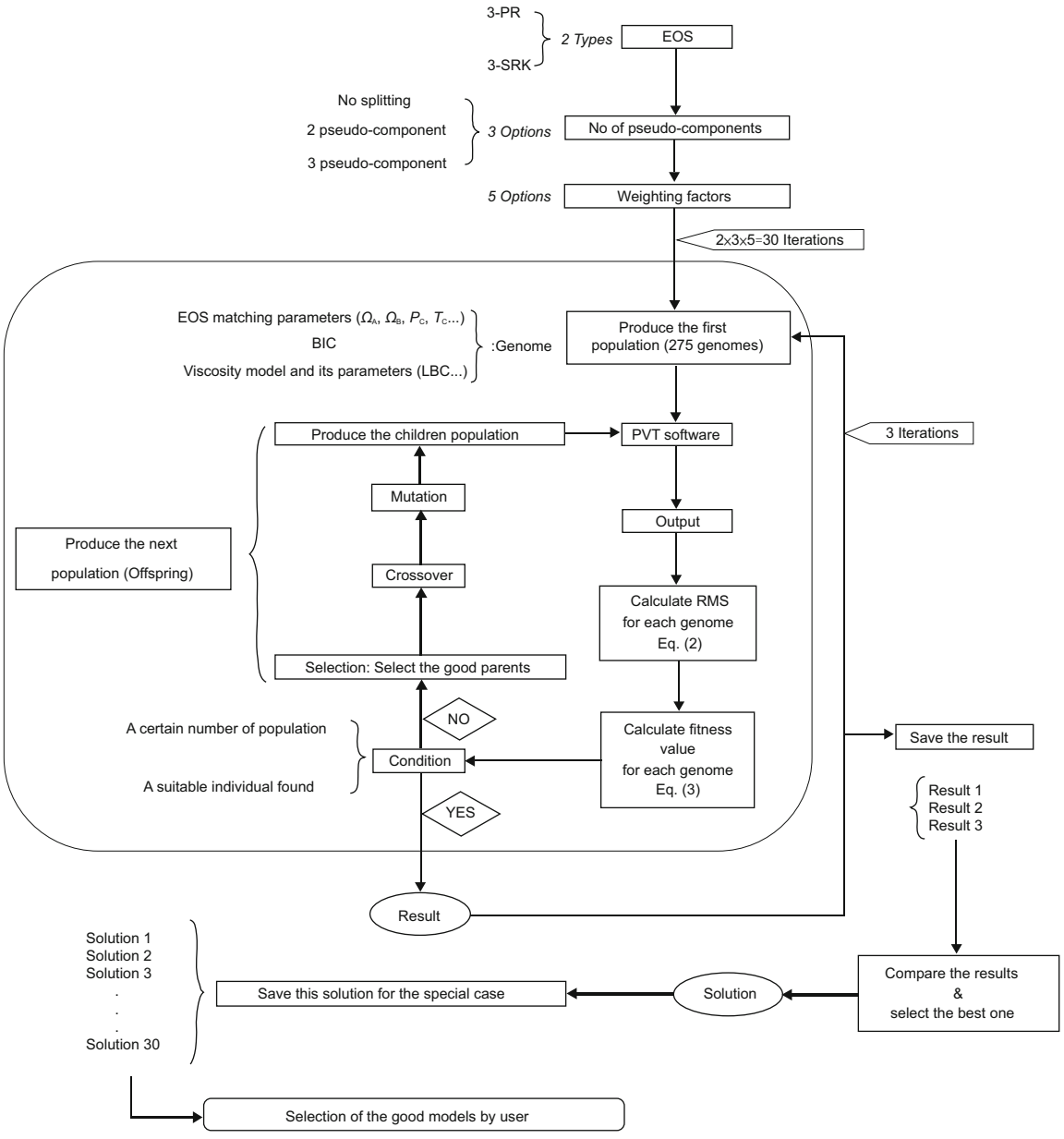

Fig. 2 Simplified diagram of the process

for the GA that we have used in this study.

3.1 Initialization

The information that is to be held within the genome is the EOS and viscosity model matching parameters, BIC, and the models of viscosity. Since the selected EOS is either 3-PR or 3-SRK, the matching parameters are Ω_a, Ω_b, P_c, T_c, acentric factor and volume shift. Also the most widely used viscosity models, namely Lohrenz-Bray-Clark (LBC), Pedersen, and Aasberg-Petersen are used in this work. So, V_c and Z_c are also added to the matching parameters for the LBC correlation if this model is selected for modeling viscosity.

Now we need to specify the structure of the genome. In spite of the fact that the general structure for genome is a one-dimensional array with each number specified by binary bits, we have chosen to use a non-standard structure for the genome. The variables have been split into three groups with each group being allocated to a separate chromosome. The chromosome for the EOS parameters is a two-dimensional array and forms an 8*NCOMPS array of binary numbers. NCOMPS shows the number of components of the fluid sample. Also, the chromosome for BIC is a two-dimensional array and forms a NCOMPS*NCOMPS array of binary numbers and the chromosome of the models of viscosity is a one-dimensional array of binary numbers with three bits. Fig. 3 through Fig. 5 show typical chromosomes that are used in this study. Note that binary interaction coefficient matrix is symmetric and a chromosome like Fig. 4 is a good representative of it.

In these chromosomes, 1 means that, the parameter is a regression variable in tuning of EOS and 0 means that, the parameter is not a variable. Note that using properties of the lighter components as regression variables is not a recommended method as these values are well defined, also critical volumes and Z factors are only needed for the LBC viscosity correlation. Therefore, in selecting the

regression parameters in the chromosome of the EOS and viscosity models parameters (Fig. 3), the program assigned a probability number to each block following general rules. For example, the plus fraction row in the matrix has the highest probability, the critical volume and *Z* factor columns have zero probability when the selected viscosity model is not LBC, or the probability of the lighter components rows are zero.

	Omega-A	Omega-B	P_{crit}	T_{crit}	V_{crit}	Z_{crit}	Acentric-factor	*S* shift
H_2S	0	0	0	1	0	0	1	0
C_1	0	0	0	0	0	0	0	0
C_2	0	0	0	0	0	0	0	0
C_3	0	0	0	0	0	0	0	0
C_4	0	0	0	0	1	0	1	0
C_6	0	1	0	1	1	0	1	0
C_{7+}	1	1	1	1	0	1	0	1

Fig. 3 Example of a two-dimensional chromosome for the EOS parameters

	H_2S	C_1	C_2	C_3	C_4	C_6	C_{7+}
H_2S							
C_1	0						
C_2	1	1					
C_3	1	0	0				
C_4	0	0	0	0			
C_6	0	1	0	0	1		
C_{7+}	1	0	1	1	0	0	

Fig. 4 Example of a two-dimensional chromosome for the BIC

Lohrenz-Bray-Clark	Pedersen	Aasberg-petersen
1	0	0

Fig. 5 Example of a one-dimensional chromosome for the viscosity models

Considering these limitations, the program will produce the parent population.

3.2 Selection

When the parent population is initialized, solutions from this population are taken and used to form a new population (offspring). This is motivated by a hope, that the new population will be better than the old one. Solutions which are then selected to form new solutions are selected according to their fitness; the more suitable they are, the more chances they have to reproduce. So, the first step is to calculate the fitness value of each individual.

In this study, the following root mean square (RMS) equation is used as an objective function:

$$RMS = \sqrt{\frac{1}{N}\sum_{i=1}^{N}\left[W_i \times \left(\frac{X_i - x_i}{x_i}\right)\right]^2} \tag{2}$$

where N represents the total number of experimental points; W_i represents the weighting factor for each point; X_i represents the calculated value; and x_i represents the experimental value.

Note that, the smaller the RMS value, the better the chromosome is. So, to simplify the problem, the output of this objective function is used to calculate fitness value by the following fitness function:

$$f_i = \frac{1 - RMS_i}{\sum_{j=1}^{M}(1 - RMS_j)} \tag{3}$$

where f_i represents the fitness value of the individual i; M represents the number of individuals in the population; and RMS_i represents the output of the objective function for the individual i.

As you can see, the fitness function is a relative function and therefore, the fitness values are between 0 and 1. Also, the greater the fitness value, the better the chromosome is.

When the fitness of each individual is specified, the selection operator chooses better individuals according to their fitness values. The selection method used in this study is a stochastic sampling method called "roulette wheel" (Shopova and Vaklieva-Bancheva, 2006).

This method is the simplest proportionate selection scheme. In this method, parents are selected according to their fitness. The better the chromosomes are, the more chances to be selected they have. In this method, the individuals of the population assume as slots of the Roulette-wheel. Each slot is as wide as the probability for selection of the corresponded chromosome is great. The size of the slot in the roulette wheel is proportional to the value of the fitness function of every chromosome. Fig. 6 shows the principle of this selection method.

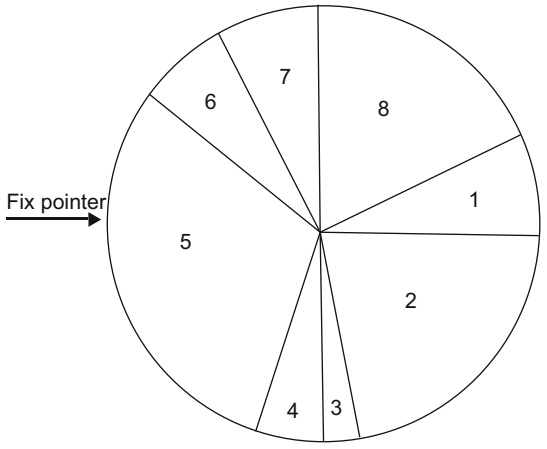

Fig. 6 Roulette wheel selection

The wheel is rotated and the chromosome that stops in front of a fixed pointer is selected. Clearly, the chromosomes with a larger fitness value will be selected more often. Roulette-wheel selection gives preference to the better individuals in the population and exerts a large pressure on the search process.

This process can be described by the following algorithm:

1) [Sum] Calculate the sum of all chromosome fitness in population (S).

2) [Select] Generate a random number from the interval (0, S).

3) [Loop] Go through the population and sum the fitness value of each chromosome in a sequential process. When the sum is greater than the random number, stop the summation process and select the previous chromosome.

This loop is iterated while the required numbers of chromosomes are selected.

3.3 Crossover and mutation operators

The selected parents should produce the next population by crossover and mutation operators. Because of the special genome structure in this study, we use specially designed crossover and mutation operators.

The crossover adopted here is a kind of 2-point crossover technique. In this method, two random numbers are produced. Then, for crossover in the two-dimensional chromosomes of EOS and viscosity model parameters, the *x*-coordinates of these numbers are normalized between 1 and 8, and the *y*-coordinates of these numbers are normalized between 1 and the number of components of the fluid sample (NCOMPS). Also, for crossover in the two-dimensional chromosomes of BIC, the *x* and *y*-coordinates of these numbers are normalized between 1 and NCOMPS. The coordinates of these numbers, show the crossover region in the two-dimensional chromosome. Fig. 7 shows different situations for crossover in two-dimensional chromosomes according to *x* and *y*-coordinates of the random numbers.

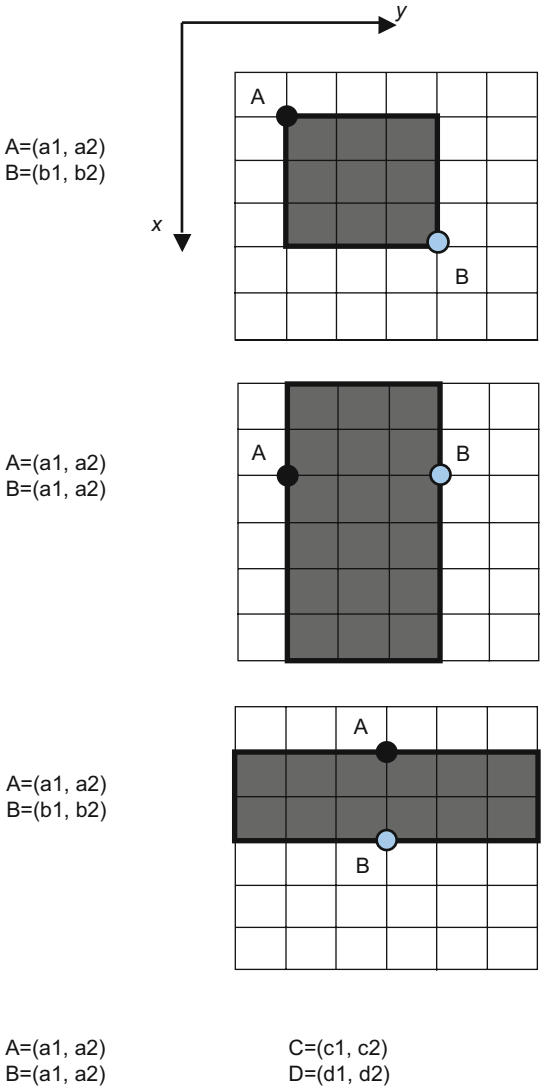

Fig. 7 Examples of crossover in two-dimensional chromosomes

For the one-dimensional chromosomes of viscosity models, the crossover is meaningless since just one block of this chromosome should be selected that shows the method of modeling viscosity. We cannot select 2 models simultaneously nor select none. So, this operator will not perform on these chromosomes and they will be repeated in the next population with no crossover.

The mutation method used in this study is a kind of one point mutation technique. In this method, one random number is produced. Then, like the crossover operator, the coordinates of the random number are normalized. These coordinates specify the mutation point like the crossover operation and flip (0 to 1, or 1 to 0) the gene in the two-dimensional chromosomes.

For the viscosity model chromosome, the program generates a random number between 0 and 1. If the random number is less than 0.33, the program will flip the first block and if the number is between 0.33 and 0.67, flip the second block, else the third block will be flipped. In these chromosomes, if the value of the selected block for mutation is 0, the value will change to 1 and the values of the other blocks change to 0, but if the value of the selected block for mutation is 1, no change will be occurred. This is because only one viscosity model should be selected.

Both the crossover and mutation operators are performed on the entire population, except viscosity models chromosomes, with different probabilities. It means that, an individual may suffer crossover, mutation, or both, or no change with respect to its generated random number. For example, if the crossover probability is *C* and the mutation probability is *M*, first we generate a random number for each individual. If the random number is smaller than *C* and *M*, crossover and mutation are performed on the individual.

When the offspring population is produced, we compare all individuals of parent and offspring populations and replace the better individuals in the new population until the new population is completed. According to the study under consideration, this method causes the program to converge fast. Also, using this method inserts an elitism operator into the program automatically.

In this study, some rules of thumb are used to determine the GA parameters. The population size is 275 individuals that is equal to the number of variables, the number of generations is 100, the crossover probability is 0.6, and the mutation probability is 0.004 that is equal to 1/variable number (De Jong, 1975; Goldberg, 1989).

4 Sample data

The fluids studied were three real black oil samples from Iranian oil fields. The compositions of these fluids are listed in Table 1. Note that as a first step, before any regression is considered, the consistency and quality of the measured data are checked.

The data that have been used in tuning of EOS are from two experiments performed on the fluids: constant composition expansion (CCE) and differential liberation (DL). Table 2 to Table 5 show the summaries and results of these tests.

Table 1 Composition of the samples

Components	Value, mol%		
	Black oil-1	Black oil-2	Black oil-3
H_2S	1.39	0.00	1.45
N_2	0.88	0.36	0.23
CO_2	5.18	0.51	3.84
C_1	22.57	25.24	26.03
C_2	6.94	7.91	7.81
C_3	5.91	5.48	5.67
i-C_4	0.98	1.07	1.01
n-C_4	2.97	3.39	3.09
i-C_5	0.93	1.42	1.02
n-C_5	1.03	1.73	1.15
C_6	3.07	4.98	2.73
C_7	4.06	2.60	6.16
C_8	4.14	1.49	3.69
C_9	3.69	2.76	2.77
C_{10}	3.45	3.10	2.75
C_{11}	2.11	2.44	3.28
C_{12+}	30.68	35.52	27.30
Total	100.00	100.00	100.00

Table 2 Summary of CCE and DL tests

Parameter	Value		
	Black oil-1	Black oil-2	Black oil-3
CCE test			
Saturation pressure, psia	1852	1404	2014
Solution GOR, SCF/STB	552	366.30	498.08
Oil gravity of residual oil, °API	24.35	24.46	20.71
DL test			
Test temperature, °F	255	220	255
Saturation pressure, psia	1845	1719	2013
Solution GOR, SCF/STB	547	420.19	498.03
Formation volume factor @ saturation pressure B_o, RB/STB	1.457	1.332	1.405
Oil gravity of residual oil, °API	24.46	22.54	20.88

Notes: GOR: Gas oil ratio; SCF/STB: Standard cubic feet per stock tank barrel; RB/SCF: Reservoir barrel per stock tank barrel.

Table 3 Results of the DL test for black oil-1

Pressure psia	Oil density g/cm^3	B_o RB/STB	Solution GOR SCF/STB	Gas compressibility factor	Oil viscosity cp
6044	0.760	1.395			1.335
5041	0.753	1.408			1.286
4043	0.746	1.421			1.237
3532	0.742	1.428			1.212
3030	0.738	1.436			1.188
2523	0.734	1.444			1.163
2323	0.732	1.448			1.153
2223	0.731	1.450			1.148
2123	0.730	1.452			1.144
2023	0.730	1.453			1.139
1925	0.729	1.455			1.134
1845	0.728	1.457	547.10		1.129
1521	0.734	1.427	478.08	0.875	1.146
1223	0.741	1.394	406.76	0.884	1.196
921	0.750	1.358	334.26	0.897	1.274
621	0.760	1.320	259.84	0.915	1.370
322	0.770	1.274	172.76	0.935	1.500
14.7	0.841	1.078	0.00	1.000	2.161

Table 4 Results of the DL test for black oil-2

Pressure psia	Oil density g/cm^3	B_o RB/STB	Solution GOR SCF/STB	Gas compressibility factor	Pressure psia	Oil viscosity cp
4992	0.792	1.293			4986	1.313
4493	0.789	1.298			3989	1.232
3995	0.786	1.303			2993	1.150
3498	0.783	1.309			2493	1.109
3001	0.779	1.315			1995	1.068
2503	0.776	1.321			1719	1.047
2206	0.773	1.325			1473	1.056
2107	0.772	1.326			1204	1.090
2008	0.772	1.327			905	1.204
1909	0.771	1.329			605	1.376
1811	0.770	1.330			305	1.570
1719	0.769	1.332	420.19		14.7	2.611
1513	0.773	1.316	381.66	0.873		
1263	0.780	1.295	336.43	0.889		
1013	0.786	1.275	291.02	0.900		
763	0.796	1.250	245.40	0.91		
513	0.802	1.229	196.32	0.928		
263	0.812	1.198	140.05	0.946		
14.7	0.864	1.062	0.00	1.000		

Table 5 Results of the DL test for black oil-3

Pressure psia	Oil density g/cm^3	B_o RB/STB	Solution GOR SCF/STB	Gas compressibility factor	Oil viscosity cp
5016	0.781	1.363			1.722
4021	0.774	1.376			1.634
3019	0.766	1.390			1.546
2617	0.763	1.395			1.510
2516	0.762	1.397			1.501
2416	0.761	1.398			1.492
2315	0.760	1.400			1.483
2215	0.760	1.402			1.475
2114	0.759	1.403			1.466
2013	0.758	1.405	498.03		1.457
1614	0.766	1.373	428.80	0.885	1.484
1212	0.776	1.337	350.21	0.897	1.626
814	0.786	1.300	270.92	0.913	1.876
411	0.797	1.258	183.64	0.937	2.238
14.7	0.863	1.075	0.00	1.000	4.288

5 Results and discussion

By running the program for the black oil samples, 30 answers are produced for each sample. For black oil-1, six answers, for black oil-2, seven answers and for black oil-3, nine answers were acceptable as engineering aspects. Fig. 8 to Fig. 10 show the results of the tuning of EOS against measured data. As the figures show, the selected models predict the fluid behavior well. Also, Table 6 shows the results for saturation pressure and oil formation volume factor at saturation pressure.

Table 6 Saturation pressure and oil formation volume factor at saturation pressure for the DL test

Parameter	Measured value	Calculated value
Black oil-1		
Saturation pressure, psia	1845.00	1844.95
Oil formation volume factor @ saturation pressure, RB/STB	1.457	1.460
Black oil-2		
Saturation pressure, psia	1719	1718.752
Oil formation volume factor @ saturation pressure, RB /STB	1.332	1.328
Black oil-3		
Saturation pressure, psia	2013	2012.641
Oil formation volume factor @ saturation pressure, RB /STB	1.405	1.408

To analyze the results more efficiently, error values were calculated for different properties. Table 7 shows the results of this analysis. In calculating error values, the following equation was used:

$$ERROR = \frac{1}{N}\sum_{i=1}^{N}\left[\left(\frac{X_i - x_i}{x_i}\right)\times 100\right] \quad (4)$$

where N represents the total number of experimental points for the property; X_i is the calculated value for the property; and x_i is the measured value for the property.

Table 7 Error values in the determination of various parameters

Parameter	Error value, %		
	Black oil-1	Black oil-2	Black oil-3
Saturation pressure	0.003	0.014	0.018
Formation volume factor @ saturation pressure	0.197	0.304	0.206
Relative volume	1.447	0.339	0.194
Oil density	0.571	0.605	0.422
Oil formation volume factor	0.543	0.842	0.381
Oil viscosity	5.548	4.245	5.832
GOR	2.293	3.912	1.105
Z factor of gas	0.340	0.368	0.613
Average error value	1.368	1.329	1.096

Also, Table 8 shows the values of matching parameters and Table 9 shows the values of BICs as a symmetric matrix for fluid sample-1 after tuning of the model.

Results show that the presented method performed well in tuning the EOS. The error values show that the GA operation is really surprising in all cases. The main advantage of our algorithm is its high speed in finding the solution. While tuning of EOS is a tedious and difficult work even for an experienced reservoir engineer and often needs a long time, the proposed method can find the solution, by confining the supervision by an expert to the last stage, in a short time and save time and expenditure in reservoir studies. Furthermore, this method prepares several possible answers for a problem simultaneously. In the manual tuning of EOS, we could find only one solution after a long time, but in this method, the algorithm tests 30 different situations and delivers the best answers at the end of each evaluation. Therefore, one might test several alternatives in a short time and choose the desired answers. Also, it is possible to use these different solutions as a sensitivity analysis. Although the figures and tables show the best answer of GA, the other physically accepted answers of the algorithm are satisfying for a reservoir study. As stated earlier, at the end of the program, there were 30 good answers and we selected some of them. Deficiency in the other answers could be related to weighting factors. Weighting factors play an important role in tuning of EOS. In this method, weighting factors are specified at the start of the

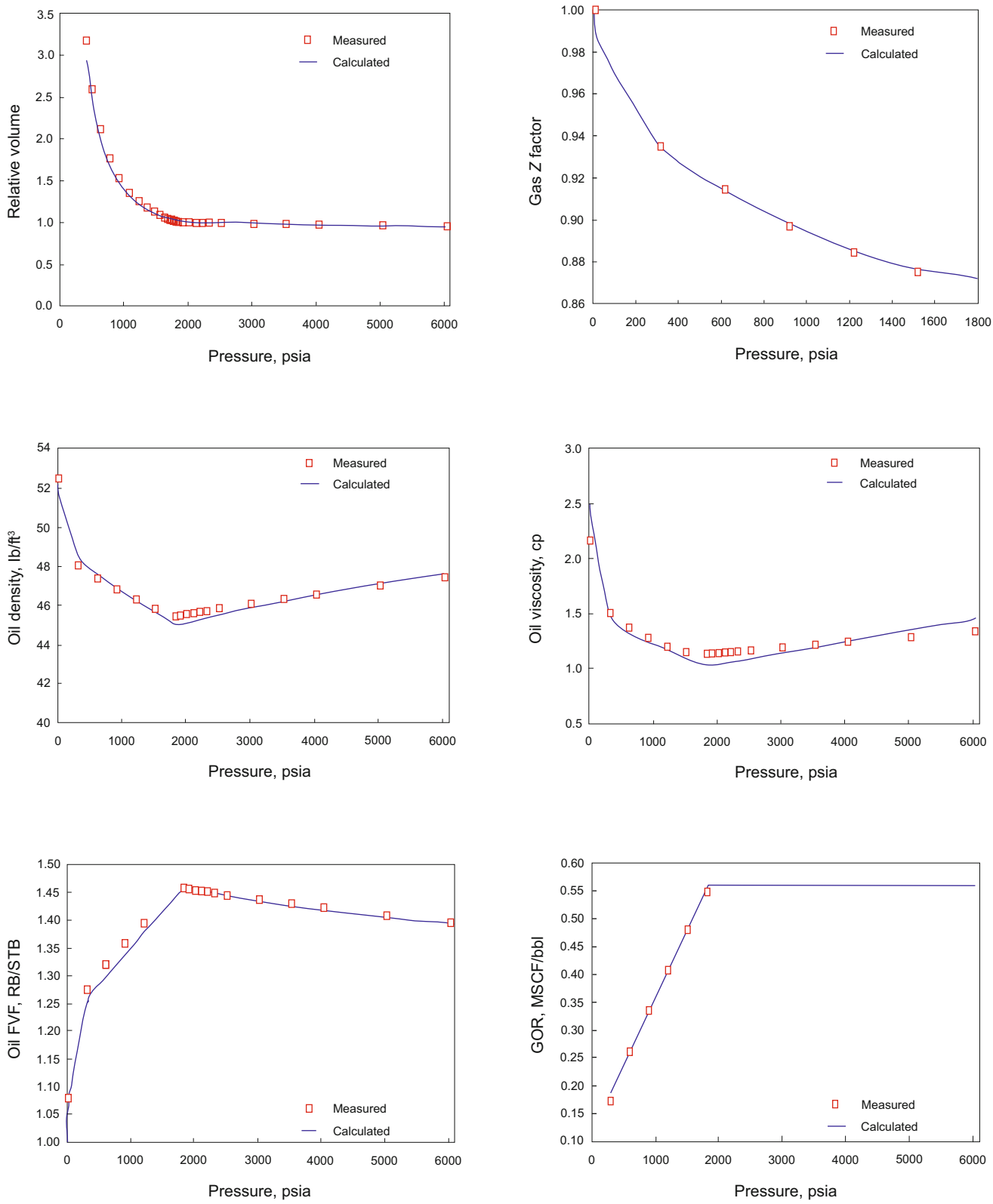

Fig. 8 Results of the tuning of EOS for black oil-1

program and are fixed till the end of one evaluation while in manual tuning these factors can change through the process by the user. Hence, by improving the program in this way, we may expect to have most of the 30 good answers desirable.

Note that tuning of EOS is not a unique work and a fluid sample could be modeled with several different tuned models. Furthermore, it is possible that a tuned model is achieved in different ways. Since the GA is a stochastic algorithm and the five groups of weighting factors are produced randomly for each case, we would expect different results and tuned

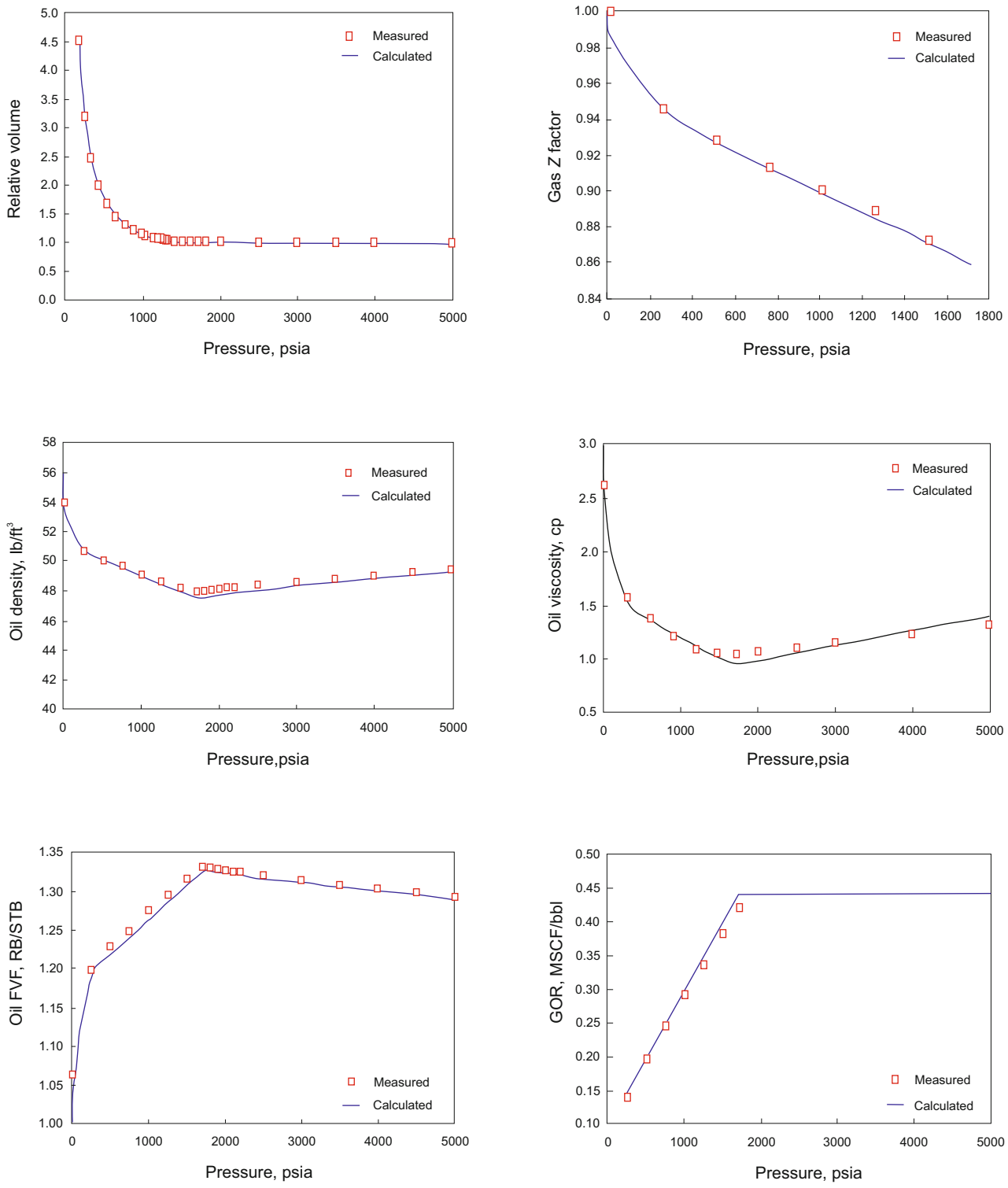

Fig. 9 Results of the tuning of EOS for black oil-2

models for each case if we re-run the program. To evaluate the variation caused by the stochastic algorithm, the program is run 10 times for black oil-1. Fig. 11 shows the number of acceptable solutions as engineering aspects for 10 different runs. Also, Fig. 12 shows the RMS and average error values for the best solution of each run. Since the RMS defined by Eq. (2) depends on the weighting factors and these factors change in each run, the following equation is used in calculating the RMS (RMS*) values to compare the results:

$$RMS^* = \sqrt{\frac{1}{N}\sum_{i=1}^{N}\left(\frac{X_i - x_i}{x_i}\right)^2} \quad (5)$$

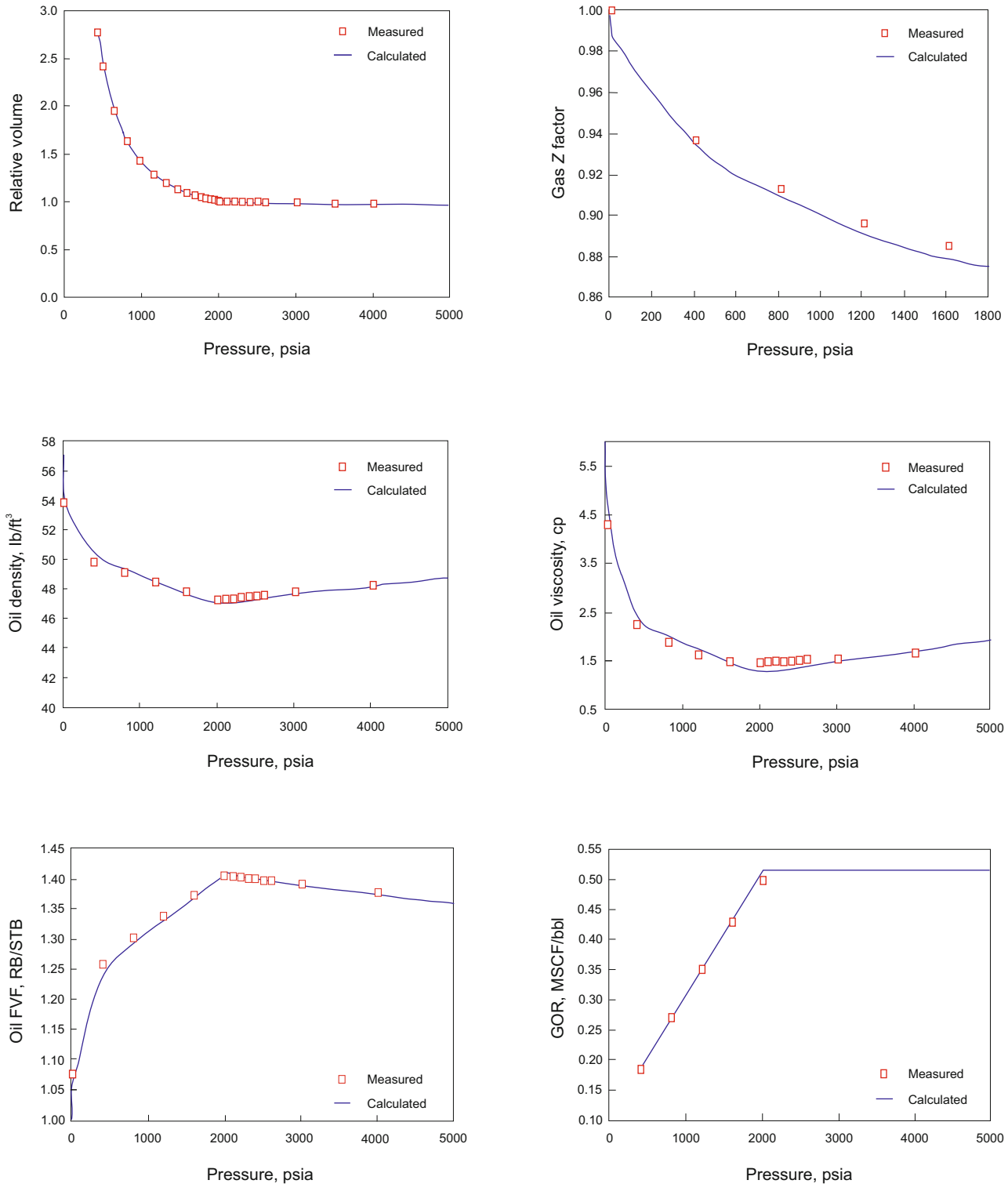

Fig. 10 Results of the tuning of EOS for black oil-3

Since the GA is a stochastic algorithm, different results are achieved for different runs of the program and no especial relation could be found among the results in the above graphs. Since both the RMS* and average error values depend on the difference between the observed and calculated values, their trends are almost the same. Also, the above graphs show that the proposed method could find the solution in each run.

These advantages introduce the proposed algorithm as a suitable method in tuning EOS against PVT experimental data for ordinary black oils.

Table 8 Values of matching parameters for black oil-1

Components	Omega-A	Omega-B	p_{crit} psia	T_{crit} °F	V_{crit} (V_{isc}) $ft^3/(lb \cdot mol)$	Z_{crit} (V_{isc})	Acentric factor	*S* shift
H_2S	0.45724	0.077796	1296.2	212.81	1.5698	0.28195	0.1	-0.1025978
N_2	0.45724	0.077796	492.31	-232.51	1.4417	0.29115	0.04	-0.1313342
CO_2	0.45724	0.077796	1071.3	88.79	1.5057	0.27408	0.225	-0.0427303
C_1	0.45724	0.077796	667.78	-116.59	1.5698	0.28473	0.013	-0.1442656
C_2	0.45724	0.077796	708.34	90.104	2.3707	0.28463	0.0986	-0.1032684
C_3	0.45724	0.077796	615.76	205.97	3.2037	0.27616	0.1524	-0.0775014
i-C_4	0.35598	0.060789	264.53	274.91	4.2129	0.14137	0.1848	-0.0619837
n-C_4	0.32196	0.070838	275.33	305.69	4.0847	0.13693	0.201	-0.0542249
i-C_5	0.45724	0.077796	491.58	369.05	4.9337	0.27271	0.227	-0.0417725
n-C_5	0.45724	0.077796	488.79	385.61	4.9817	0.26844	0.251	-0.0302779
C_6	0.45724	0.077796	436.62	453.83	5.6225	0.25042	0.299	-0.0072888
C_7	0.45724	0.077796	426.18	526.73	6.2792	0.25281	0.3	0.0575821
C_8	0.45724	0.077796	417.66	575.33	6.936	0.26082	0.312	0.031934
C_9	0.45724	0.077796	381.51	625.73	7.7529	0.25394	0.348	0.0594578
C_{10}	0.61738	0.054329	175.47	667.13	8.5539	0.12413	0.385	0.0861113
C_{11}	0.32196	0.070838	161.73	706.73	9.4028	0.12149	0.419	0.1139716
C_{12+}	0.42403	0.056428	43.473	1122.7	22.792	0.058352	2.2726	0.766346

Table 9 Values of BICs for black oil-1

Component	H_2S	N_2	CO_2	C_1	C_2	C_3	*i*-C_4	*n*-C_4	*i*-C_5	*n*-C_5	C_6	C_7	C_8	C_9	C_{10}	C_{11}	C_{12+}
H_2S	0																
N_2	0.176	0															
CO_2	0.096	-0.012	0														
C_1	0.05	0.1	0.1	0													
C_2	0.05	0.1	0.1	0	0												
C_3	0.05	0.1	0.1	0	0	0											
i-C_4	0.05	0.1	0.1	0	0	0	0										
n-C_4	0.05	0.1	0.1	0	0	0	0	0									
i-C_5	0.05	0.1	0.1	0	0	0	0	0	0								
n-C_5	0.05	0.1	0.1	0	0	0	0	0	0	0							
C_6	0.05	0.1	0.1	0.0279	0.01	0.01	0	0	0	0	0						
C_7	0.05	0.1	0.1	0.03308	0.01	0.01	0	0	0	0	0	0					
C_8	0.05	0.1	0.1	0.0363	0.01	0.01	0	0	0	0	0	0	0				
C_9	0.05	0.1	0.1	0.03896	0.01	0.01	0	0	0	0	0	0	0	0			
C_{10}	0.05	0.1	0.1	0.04092	0.01	0.01	0	0	0	0	0	0	0	0	0		
C_{11}	0.05	0.1	0.1	0.04246	0.01	0.01	0	0	0	0	0	0	0	0	0	0	
C_{12+}	0.0814079	0.1	0.162816	0.06297	0.01	0.01	0	0	0	0	0	0	0	0	0	0	0

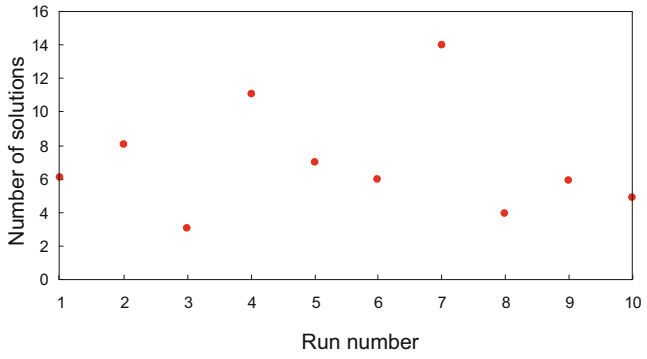

Fig. 11 Number of physically accepted solutions for 10 different runs

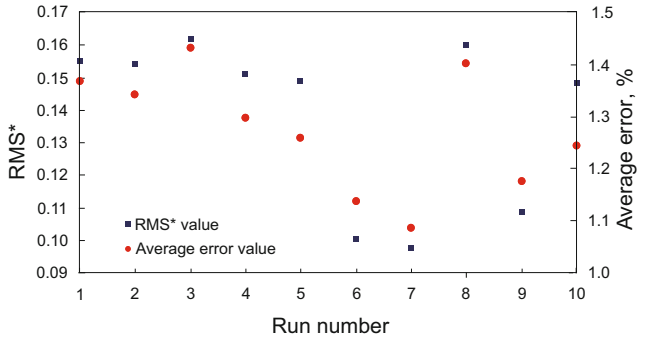

Fig. 12 RMS and average error values for 10 different runs

6 Conclusions

The results of this study show that the developed genetic algorithm can be successfully applied to the tedious, difficult, and time consuming operations of tuning of EOS against experimental data for three real black oil fluid samples. The property graphs show the successful tuning of EOS against measured data; furthermore, the average error values are below 2 percent for all the cases and prove that the GA operation is really surprising in all cases. However, at present, we cannot argue that the method presented in this study is the best form of the genetic algorithm for this problem.

The strong non-linearity of the EOS tuning process makes classical deterministic optimization methods inefficient and unlikely to be successful. Therefore, an alternative approach would be to use heuristic type methods like genetic algorithms. The ability of genetic algorithm to use continuous and discontinuous variables, changing several variables simultaneously and the ability of this method to work with different data structures in the same time, cause this optimization method to be a good choice to solve the problem of matching PVT data automatically.

The main advantage of the method is its high speed in finding solutions. While tuning the EOS is tedious and difficult work even for an experienced reservoir engineer and often needs a long time to find just one tuned model, the proposed method can find more than one solution in a short time. Also, working automatically, confining the role of experts to the last stage, reducing costs and having the possibility of evaluating the different situations are the other advantages of this method to match PVT data and makes it an ideal method to implement as an automatic EOS tuning algorithm for ordinary black oils.

Comparing the RMS values for different iterations shows that modifying the matching parameters indiscriminately does not develop the model necessarily and may results in the models that are mathematically and physically unacceptable.

Since the GA is a stochastic algorithm, different results are achieved for different runs of the program and the proposed method was successful in finding the solution in each run.

References

Bunnag D and Sun M. Genetic algorithm for constrained global optimization in continuous variables. Applied Mathematic Computation. 2005. 171(1): 604-636

Coats K H and Smart G T. Application of a regression-based EOS PVT program to laboratory data. SPE Reservoir Engineering. 1986. 1(3): 277-299

Danesh A. PVT and Phase Behaviour of Petroleum Reservoir Fluids (lst Edition). Elsevier Scientific Publishing Company. 1998

De Jong K A. An analysis of the behavior of a class of genetic adaptive systems. Ph.D Thesis. University of Michigan, Dissertation Abstracts International. 1975. 36(10): 514B

Fogarty T C, Vavak F and Cheny P. Use of the genetic algorithm for load balancing of sugar beet presses. In: Eshelman L J (Eds.), Proceedings 6th International Conference on Genetic Algorithms. Morgan Kaufmann, San Francisco, 617-624. 1985

Forrest H B, Koza J R, Shipman J, et al. Building a parallel computer system for $18,000 that performs a half peta-flop per day. Proceedings of the Genetic and Evolutionary Computation Conference. Morgan Kaufmann, San Francisco, 1484-1490. 1999

Goldberg D E. Computer-aided gas pipeline operation using genetic algorithms and rule learning. Ph.D. Dissertation. The University of Michigan Press, Ann Arbor, MI. 1983

Goldberg D E. Genetic Algorithms in Search, Optimization, and Machine Learning. Addison-Wesley Publication Company, MA. 1989

Gretenstette J, Gopal R, Rosmaita B, et al. Genetic algorithms for the travelling salesman problem. In: Gretenstette J (Eds.), Proceedings First International Conference on Genetic Algorithms and Their Applications. Lawrence Erlbaum Associates Publisher, NJ, 160-168. 1985

Holland J H. Adaptation in Natural and Artificial Systems: An Introductory Analysis with Applications to Biology, Control, and Artificial Intelligence. The University of Michigan Press, Ann Arbor, MI. 1975

Konak A, Coit D W, Smith A E. Multi-objective optimization using genetic algorithms: A tutorial. Reliability Engineering & System Safety. 2006. 91(9): 992-1007

Michalewicz Z. Genetic Algorithm+Data Structure=Evolution Programs (2nd Edition). Springer-Verlag, NY. 1992

Nikravesh M. Aminzadeh F and Zadeh L A. Soft Computing and Intelligent Data Analysis in Oil Exploration (1st Edition). Elsevier Scientific Publishing Company. 2003

Parnee I C and Watson A H. Preliminary airframe design using co-evolutionary multiobjective genetic algorithms. In: Banzhat W. Daida J, Eiben A E, et al (Eds.), Proceedings of the Genetic and Evolutionary Computation Conference 1999. Morgan Kaufmann, San Francisco, 1657-1665. 1999

Poon P W and Parks G T. Optimizing PWR reload core designs. In: Manner R and Manderich B (Eds.), Parallel Problem Solving from Nature 2. North-Holland, 371. 1992

Romero C E and Carter J N. Using genetic algorithms for reservoir characterization. Journal of Petroleum Science and Engineering. 2001. 31(2-4): 113-123

Shopova E G and Vaklieva-Bancheva N G. BASIC-A genetic algorithm for engineering problems solution. Computers and Chemical Engineering. 2006. 30: 1293-1309

Velez-Langs O. Genetic algorithms in oil industry: An overview. Journal of Petroleum Science and Engineering. 2005. 47(1-2): 15-22

(Edited by Sun Yanhua)