The Pennsylvania State University

The Graduate School

John and Willie Leone Family Department of Energy and Mineral Engineering

RATE TRANSIENT ANALYSIS OF DUAL LATERAL WELLS IN NATURALLY FRACTURED RESERVOIRS VIA ARTIFICIAL INTELLIGENCE

A Thesis in

Energy and Mineral Engineering

by

Jia Lu

© 2015 Jia Lu

Submitted in Partial Fulfillment

of the Requirements

for the Degree of

Master of Science

May 2015

The thesis of Jia Lu was reviewed and approved* by the following:

Turgay Ertekin

Head, John and Willie Leone Family Department of Energy and Mineral Engineering
Professor of Petroleum and Natural Gas Engineering
George E. Trimble Chair in Earth and Mineral Sciences
Undergraduate Program Officer of Petroleum and Natural Gas Engineering
Thesis Advisor

Zuleima T. Karpyn

Associate Professor of Petroleum and Natural Gas Engineering
Quentin E. and Louise L. Wood Faculty Fellow in Petroleum and Natural Gas Engineering

Li Li

Assistant Professor of Petroleum and Natural Gas Engineering

Luis F. Ayala H

Associate Professor of Petroleum and Natural Gas Engineering
Associate Department Head for Graduate Education

*Signatures are on file in the Graduate School

Abstract

In naturally fractured reservoirs, reservoir characterization is critical to the production of hydrocarbon, including but not limited to porosity, permeability, pay zone thickness and fracture spacing. Laboratory measurements, well-logging technique, and mathematical models are three major characterization approaches that are widely used to determine and analyze the reservoir characterization and production profiles. Amongst these approaches, mathematical models are commonly used as estimation tools. The purpose of this thesis is to develop a mathematical model as a reservoir estimation tool for naturally fractured reservoirs with dual lateral well configurations. The tool proposed in this study includes a forward artificial neural network (ANN) with the ability to predict production data via known reservoir and well design parameters. The proposed tool also includes an inverse ANN component that can be used to predict the permeability and porosity of matrix and fracture, as well as fracture spacing and reservoir thickness. By means of the proposed tool, the user would be able to analyze instantaneously predicted reservoir or production data with less cost and time. The software involved in developing the tool were MATLAB, EXCEL, and a commercial modeling software[1]. The procedures are introduced and discussed in the following chapters including training data generation, selecting training data sets, training forward and inverse ANN models. Moreover, a graphical user interface (GUI) is developed and assembled for each ANN, which allows the user to view results in numerical and graphical formats.

[1] The commercial modeling software used was provided by Computer Modeling Group LTD (CMG). The IMEX software was mainly used for single phase gas production.

Table of Contents

List of Figures

List of Tables

Acknowledgements

I would like to thank my thesis advisor Dr. Turgay Ertekin for being my advisor for both my undergraduate and graduate degrees from the Pennsylvania State University. I truly appreciate his erudite knowledge and illuminated assistance that guide me for my study, especially for the completion of my master thesis. I would also like to thank Dr. Zuleima T. Karpyn and Dr. Li Li for being serving on my thesis committee.

I would like to thank all my friends for their support, help, and friendship. I truly appreciate the continuous, valuable, and significant assistance from my colleagues, especially from Qian Sun and Qiumei Zhou, throughout this project.

I would like to thank all my family for their continuous support and encouragement throughout my life. I am deeply grateful to my father Mr. Yongjin Lu and my mother Mrs. Min Zhao for their vast and endless support, help, and encouragement that lead me to success.

Chapter 1: Introduction

Naturally fractured reservoirs, also called dual-porosity reservoirs, have matrix and fracture as their porous media and they can dramatically influence the hydrocarbon production. As the rapid growth of development of the petroleum industry in these decades, the demand for higher recovery efficiency and accurate production predictions have become more critical. Therefore, the analysis of reservoir characterization and production profile is brought to the forefront. For naturally fractured reservoirs, matrix porosity, fracture porosity, matrix permeability, fracture permeability, and fracture spacing are five crucial parameters that control the production of oil and gas. Porosity is normally defined as percentage of the pore space that occupies the total rock. It is a measure of its ability to store a fluid. Typically, matrix porosity is larger than fracture porosity in the naturally fractured reservoir since the matrix has the most storage volume (pore volume). Permeability is defined as the measurement of the ease a fluid can flow through the porous media. Fracture permeability is larger than matrix permeability in the naturally fractured reservoirs and that is why the channels that exist in fracture enhance the flow efficiency of fluids.

Mathematical models are constructed for estimating the reservoir. In this thesis, an artificial neural network (ANN) model was established. The model contains a forward ANN to predict the production profile and an inverse ANN to estimate reservoir characteristics of a dual-porosity reservoir.

MATLAB, EXCEL, and the commercial reservoir simulation software (CMG) are software tools utilized throughout the project. MATLAB was applied to generate initial reservoir data sets and feed into CMG, to construct ANN networks, and to finally create Graphic User Interface (GUI).

EXCEL was applied on plots and graphs, to organize and normalize data sets, and to analyze network results. CMG was used to receive the data sets from MATLAB and to generate a reservoir production profiles for each data set.

Chapter 2: Literature Review

2.1 Reservoir Characterization

Fractures, either natural or man-made, are existing in all petroleum reservoirs. Natural fractures are generated by the interactions among tectonic stresses such as shear stress, tension, and compression. On the other hand, man-made fractures are mainly created by human activities such as reservoir fluid injection, water flooding, and drilling. (“Fractured Reservoirs”, 2012). Natural fractures in reservoir formation play a critical role in hydrocarbon production and naturally fractured reservoirs constitute the primary hydrocarbon resources in the world. Up to 1956, 41% of the hydrocarbon reserves in the world are featured as fractured reservoirs according to Knebel et al. (1956) and the number of this class of reservoirs is still increasing.

Due to the significances of natural fractures and naturally fractured reservoirs, analytical models were developed to simplify and describe the fluid behavior and storage in naturally fractured reservoirs. Double-porosity model is one of the analytical reservoir which studies the formation by dividing it into fractured and unfractured domains (Streltsova, 1988). In this research, Warren and Root’s model is applied to simulate and describe the system. According to Warren and Root (1963), a physical idealization is based on the following several assumptions to simulate the behavior of a formation with single-phase flow and intermediate porosity (Warren & Root, 1963).

The assumptions for this reservoir model are:

1) Intergranular porosity consists in a homogeneous and isotropic matrix, which is an organized media with identical cuboids.

2) Fissure and/or vugular porosity is comprised in an orthogonal system of continuous uniform fractures. The fractures are paralleled to the principal axes of permeability so that the flow within fractures is also paralleled to the axes. Although the fracture width is constant along a particular axis, different fracture spacing can contribute to the anisotropy.
3) The combination of matrix porosities and fracture porosities is homogeneous and anisotropic, which determines the flow interaction from the matrix to fracture but not from the matrix to matrix. (Warren & Root, 1963).

The idealization of the heterogeneous porous medium is shown in Figure 2.1.

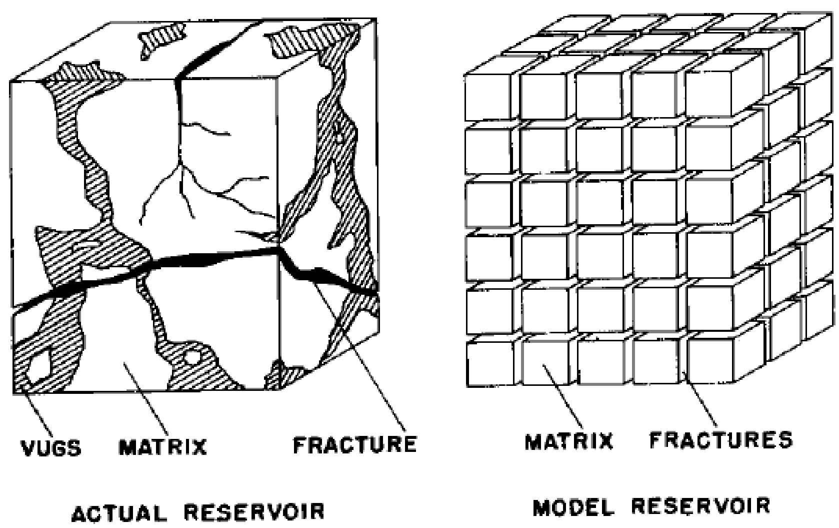


Figure 2.1 Warren & Root Double Porosity Model (Warren & Root, 1963)

According to the model, the fractures are the main conduit for fluid flow. Thus porosity, permeability, spacing and compressibility of fracture are the significant properties impact on the hydrocarbon production (Kistak, 2013). In addition, the primary porosity, which is also called intergranular porosity, has a significant role in the pore volume but negligible influence on the flow capacity (Warren & Root, 1963).

2.2 Horizontal Wells

In order to enhance reservoir performance, wells are drilled at multiple angles to better reach the target formation, which is called a horizontal well. Horizontal wells are high-angle directional wells, and this technology has been a useful tool to enhance hydrocarbon production since 1985 (Kistak, 2013). Horizontal wells can reduce water and gas coning, increase the production rate, reduce pressure drawdown of the wellbore, and increase reservoir recovery (Kistak, 2013). Figure 2.2 is a trajectory example for a horizontal well.

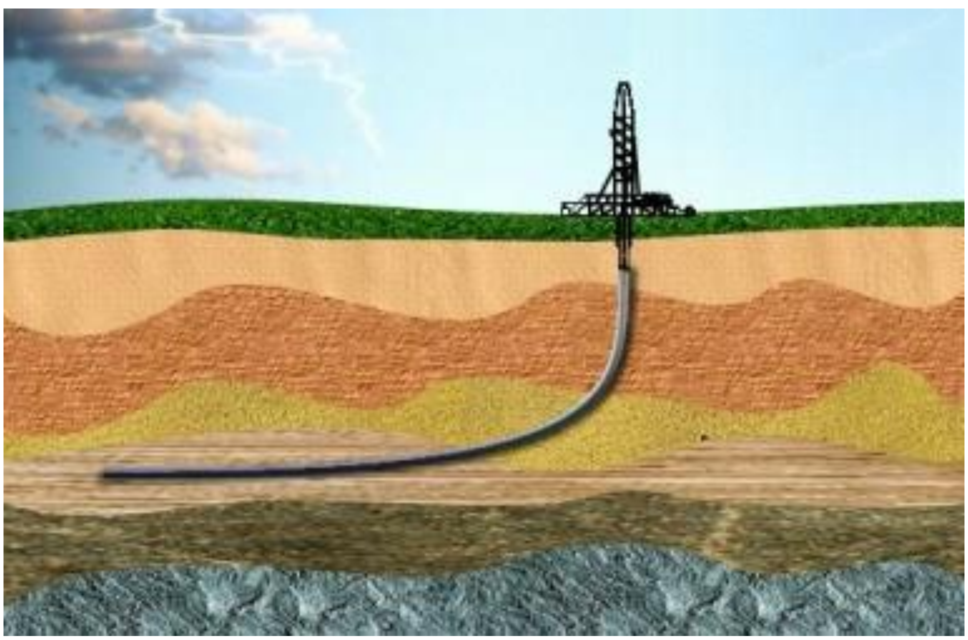


Figure 2.2 A Horizontal Well Trajectory (Swan Energy, 2012)

Multilateral wells technology is one of the new evolutions of horizontal drilling technology, with extends branches into target formation from the main wellbore. Multilateral wells can be further sub-classified into different scenarios depending on the number of the branches. Dual-lateral wells, which have two horizontal branches, were studied in this research. The illustration of a dual lateral well is shown in Figure 2.3.

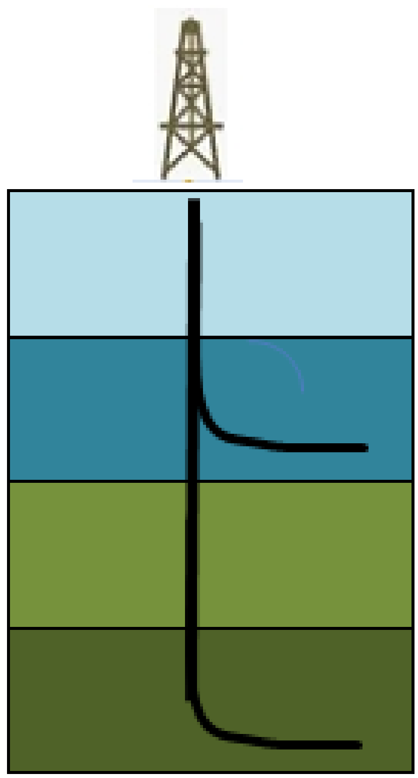


Figure 2.3 A Dual Lateral Well Configuration

Due to the displacements, phasing, and the length of the laterals, dual-lateral wells may reveal diverse flow regimes (Ozkan, 1998). Within the well pressures and pressure functions, there are at least four flow regimes that can be observed. Thses flow regimes are the early-time radial flow in the vertical plane, the intermediate time linear flow in the horizontal plane, the intermediate time pseudo-radical flow in the horizontal plane, and the late-time pseudo-radical flow in the horizontal plane. Amongst the four flow regimes, the first three can be classified into one category since two laterals do not interact with each other, and the last flow system is considered as another category since the two laterals do interfere with each other (Ozkan, 1998). The configurations of four flow regimes are presented Figure 2.4.

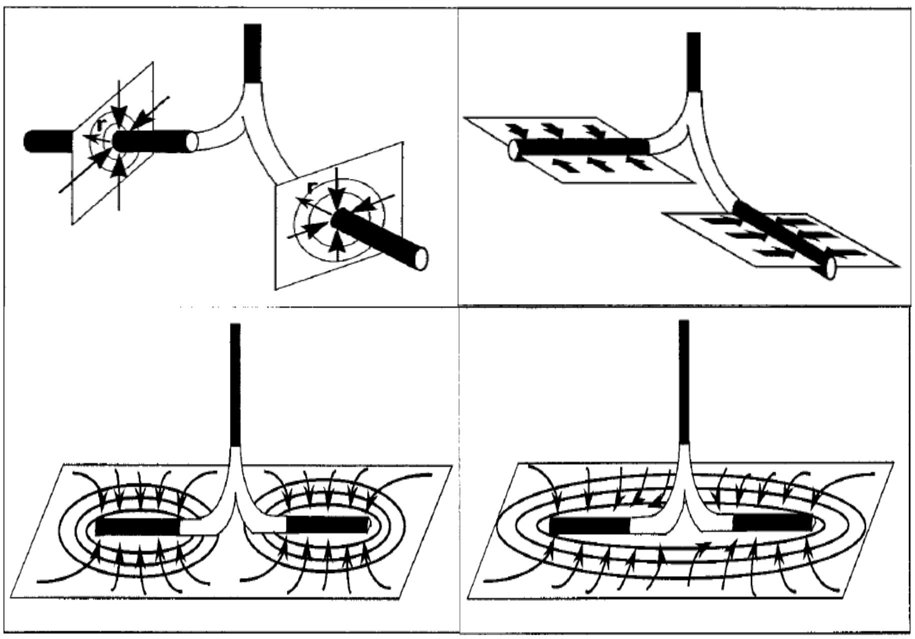


Figure 2.4 Dual-Lateral Well Flow Regimes (Ozkan, 1998)

At early times, as shown in the top left figure, each lateral has its radial flow regime perpendicular to a horizontal plane or to the axis of the lateral. The expression can be derived for the early-time radial flow regime by short-time approximation in terms of dimensionless and dimensionally groups (Ozkan, 1998):

$$P_{wD} = \frac{h_D}{2(L_{HD1}+L_{HD2})}\sqrt{\frac{k_2}{k}}\left\{\ln t_D - \ln r_{wD1}^2 + 0.80907 + \int_0^{t_D} q_{D2}(\tau)[\exp\left(-\frac{r_{wD2}^2}{4(t_D-\tau)}\right) - \exp\left(\frac{r_{wD1}^2}{4(t_D-\tau)}\right)]\frac{d\tau}{\tau}\right\} \qquad (1)$$

$$\Delta P_{wf} = \frac{162.6qB\mu}{\sqrt{k_2}L}\left\{\ln t + \ln\frac{2.637\times10^{-4}\,k}{\emptyset C_t\mu r_{w1}^2} + 0.351 + \int_0^{t_D} q_{D2}(\tau)[\exp\left(-\frac{r_{w2}^2}{4\eta(t-\tau)}\right) - \exp\left(\frac{r_{w1}^2}{4\eta(t-\tau)}\right)]\frac{d\tau}{\tau}\right\} \quad (2)$$

Two intermediate-time flow regimes may exist which are linear and pseudo-radial flow regimes. It is possible to observe intermediate-time linear flow if the laterals are long enough and intermediate time pseudo-radial flow happens when the horizontal separation between laterals is large enough (Ozkan, 1998). The expressions for both linear and pseudo-radial flows by intermediate-time approximations are shown in equations (3) and (4) below (Ozkan, 1998):

$$\Delta P_{wf} = \frac{8.13qB\sqrt{\mu}}{hL\sqrt{\emptyset C_t}}\sqrt{t} \quad (3)$$

$$\Delta P_{wf} = \frac{81.3qB\mu}{\sqrt{k_x k_y}h}\left\{\log t + \log\frac{2.637\times10^{-4}\,k}{\emptyset C_t\mu L^2} + 2.44 + \frac{2}{1.151}\sum_{i=1}^{2}\frac{q_{Di}}{L_{HDi}^2}\int_{L_{oDi}}^{L_{oDi}+L_{HDi}}(\sigma_{ii} + F_{ii})dr_D\right\} \quad (4)$$

At late times, when two lateral wells start to influence each other, flow becomes mainly pseudo-radial in the reservoir and the following expression can be derived as shown in equation (5) below:

$$\Delta P_{wf} = \frac{162.6qB\mu}{\sqrt{k_x k_y}h}\left\{\log t + \log\frac{2.637\times10^{-4}\,k}{\emptyset C_t\mu L^2} + 2.44 + \frac{2}{1.151}\sum_{i=1}^{2}\sum_{j=1}^{2}\frac{q_{Di}}{L_{HDi}L_{HDj}}\int_{L_{oDi}}^{L_{oDi}+L_{HDi}}(\sigma_{ii} + F_{ii})dr_D\right\} \quad (5)$$

2.3 Artificial Neural Network (ANN)

Artificial Neural Network (ANN), in general, is a model that can receive inputs from the previous step, transform, and pass outputs to next step. It was inspired by biological neuron system in which neurons can receive and transmit signals to each other to maintain the operation of biological systems (Demuth, Beale, & Hagan, 2009). As illustrated in Figure 2.5, a neuron contains the cell body, the dendrites, the axon, and the axon terminals. The dendrites are the connections between cells in the brain, were created when learning happens. The dendrites transform the signals to impulses and pass them to cell body (Lodish, Berk, & Zipursky, 2000). The cell body is central of the neuron that contains all the necessary biological parts to ensure the function of a neuron (Lodish, Berk, & Zipursky, 2000). Axons are the long paths for transmitting pulses from the cell body to the axon terminals (Lodish, Berk, & Zipursky, 2000). Axon terminals are responsible for connecting and passing the pulses to other cells (Lodish, Berk, & Zipursky, 2000).

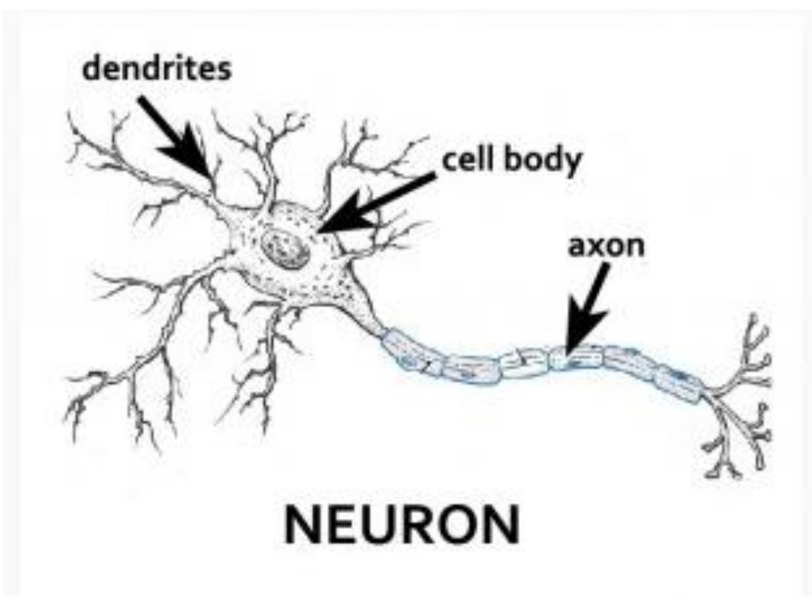


Figure 2.5 Neuron Structure (Lodish, Berk, & Zipursky, 2000)

An artificial neural network, in essence is a mathematical model which simulates the functions of a biological neural network, and is a useful tool to help to solve complicated problems quickly and accurately. The artificial neuron, a fundamental component of the ANNs, has similar functions with the biological neuron. The dendrites in ANNs are considered as input data, the cell body in ANNs has the function to link the inputs with outputs, and the axon terminals in ANNs are considered as outputs. The basic illustration of an artificial neuron can be seen in Figure 2.6. With a combination of many artificial neurons, a network can be setup to solve complicated problems by defining the relationship between inputs and outputs. The first artificial neural network was established by Frank Rosenblatt in 1950's. However, it was discovered to be a lack of accuracy since there were no learning algorithms (Enab, 2012).

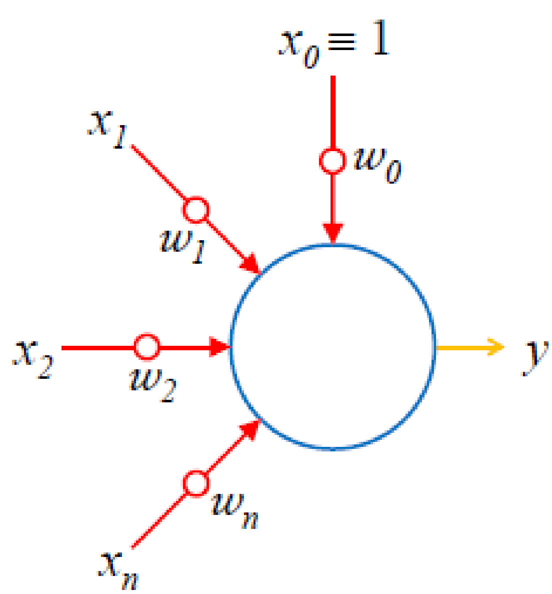


Figure 2.6 Artificial Neuron Structure (Kulga, 2010)

Two components of ANN models are utilized in this study, which are the forward solution component of the analytical model and the inverse solution component of the analytical model. With the help of an inverse ANN, desired formation characteristics were predicted such as permeability, porosity, saturation, thickness, and fracture spacing. Similarly with the help of a forward ANN, hydrocarbon production profile can be predicted by knowing reservoir and well parameters. Unsupervised and supervised protocols are two training methods used in the neural network development and the supervised learning is the most common one (Alajmi, 2007). The backpropagation neural network (BPNN) used in this work, which is one of the recent algorithms developed.

Chapter 3: Problem Statement

The rate transient well testing (RTA) is one of the efficient gas well analysis approaches. Thus, the objective of this project was to develop artificial neural networks to predict desired parameters. In this study, the forward ANN model predicts the gas production profile over a period of time of ten years, and the inverse ANN model predicts the specific reservoir characterizations via the predicted output data.

The following assumptions are in place during the development of ANNs in order to achieve this objective.

1. The produced hydrocarbon is single phase gas (single methane component)
2. The matrix and fracture characterizations are isotropic
3. The production process takes place under isothermal conditions
4. Skin factor is neglected
5. The two lateral wells are located in different depth
6. Gravitational effects are neglected
7. The surface drilling location is in the center of the reservoir
8. Two lateral wells has no phase angles

Chapter 4: Development of Artificial Neural Network

Based on the literature survey, the utilization of CMG software and the MATLAB artificial neural network (ANN) toolbox, the final ANN(s) were established and developed. In this chapter, the processes for creating ANN(s) are explained, along with the introduction of each component of network structure and the function selection criteria.

4.1 Data Preparation

The properties of reservoir and well are essential to the design process since the ANNs require a significant amount of reasonable data to establish meaningful connections between neurons. The system properties involved in this research were reservoir area, reservoir depth, reservoir thickness, matrix porosity, fracture porosity, matrix permeability, fracture permeability, fracture spacing, matrix compressibility, fracture compressibility, reservoir temperature, reservoir pressure, and bottomhole pressure. The lengths and depths of dual-later wells were selected for this research as well parameters. The parameters and their ranges are shown below in Table 4-1.

Table 4.1: Data Range for Formation and Well Properties

Reservoir Parameter	Symbol	Parameter Range		unit
		Min	Max	
Area	A	100	640	acres
Depth	D	1000	10000	ft
Thickness	H	100	400	ft
Matrix Porosity	phi_m	0.08	0.4	
Fracture Porosity	phi_f	0.001	0.03	
Matrix Permeability	k_m	0.001	0.05	mD
Fracture Permeability	k_f	0.1	1	mD
Fracture Spacing	FS	1	200	ft
Matrix Compressibility	Cm	6.00E-08	3.00E-06	1/psi
Fracture Compressibility	Cf	6.00E-07	1.00E-03	1/psi
Pressure	P	1000	8000	psi
Temperature	T	70	160	F
Bottomhole Pressure	BHP	14.7	4014.7	psi
Length of 1st Well	L1	500	2000	ft
Length of 2nd Well	L2	500	2000	ft
Depth of 1st well	D1	50	350	ft
Depth of 2nd well	D2	50	350	ft

In order to generate an expert system that is universally applicable to dual-porosity reservoirs, the properties and the ranges of parameters need to be carefully selected and discussed. There are still many constraints exist in generating the numerical reservoir models. Firstly, reservoir temperature and bottomhole pressure were determined by the depth and the initial reservoir pressure respectively. Then the well lengths need to be limited to less than half of the lengths of the edge of the drainage patterns in order to locate wellhead at the center of the reservoir. Afterwards, it is necessary to set two wells into different horizontal planes to avoid the occurrence of overlap of two lateral wells. Lastly, the depth of two lateral wells have to be equal or less than the reservoir thickness, in other words, the horizontal wells should be all located in the pay zone. Once the 2000 sets of random data were generated, 2000 reservoir models were constructed by inputting all the data sets into CMG's IMEX conventional black-oil/gas simulator.

Among the reservoir models, 35 of them were eliminated due to the invalid conditions in CMG software, thus, 1965 sets were left for further procedures. Figure 4.1 shows a 3D view of a cross section of a sample CMG reservoir model with five layers. Afterwards, a batch file was setup in MATLAB for rapidly and automatically inputting variable values into the numerical model so that 1965 sets of 10-year production files were generated. Once the output files were made, the daily field production rate were extracted from the output files, this process was also automatically controlled via another MATLAB batch file. The code for the batch files was presented in Appendix A. Production data were selected every four months to use in training process, thus, there are a total of 1965 data sets and each set contains 17 design parameters and 31 field production rate data (time series).

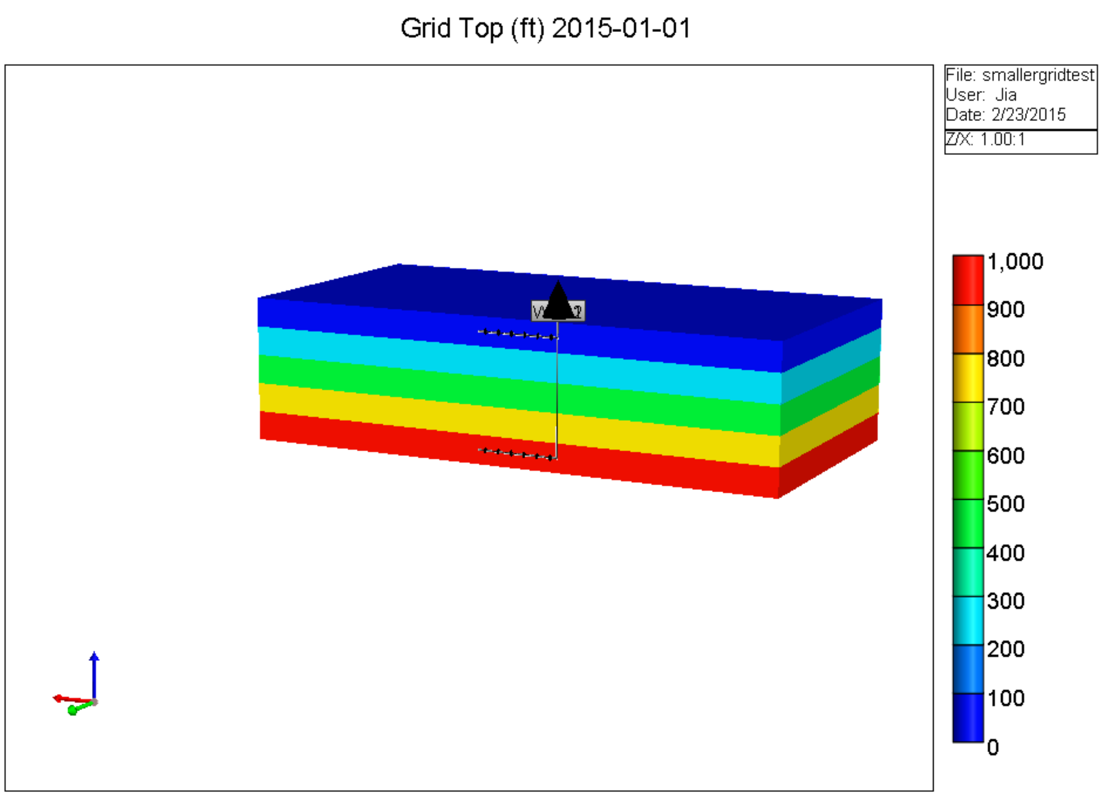


Figure 4.1 Cross Section of a Dual-Lateral Well in a Typical Reservoir Model

4.2 Network Development

An artificial neural network mainly contains four functions which are: the transfer function, the training function, the learning function, and the performance function (Kistak, 2013). The transfer function takes input from last layer to generate output to next layer (Kistak, 2013). The training algorithm modifies the weights until the output matches the target (Kistak, 2013). The learning function is used to update individual weights and biases during learning. The performance function is used to operate the artificial neural network (Kistak, 2013).

4.2.1 Architecture

The multilayer cascade feed-forward back propagation was selected as the principal architecture of the artificial neural network. It is one of the most popular types of neural network in many different types of applications. In addition, previous studies done at The Pennsylvania State University show that the back propagation algorithm gives the best training results and the most accurate network with reasonable training structures (Enab, 2012) (Kulga, 2012) (Kistak, 2013) (Alajmi, 2003) (Bansal 2009).

As shown in Figure 4.2, a typical structure of this type of ANN consists of an input layer, one or more hidden layers, and an output layer. The simple feedforward network only updates the weighting factors after the input signals have passed through the entire system. By contrast, the multilayer Cascade feed-forward back propagation network can converge faster because it can update the weighting factors according to the error-correction rule after the input signals translated to output from the hidden layers, in addition, the neurons can be adjusted and analyzed at each layer.

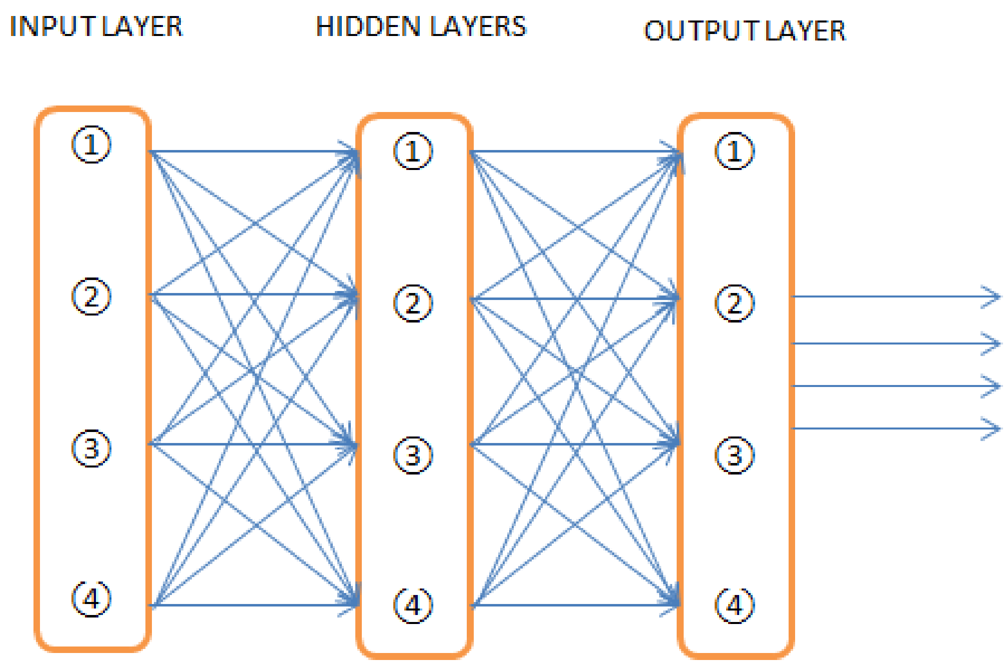


Figure 4.2 Typical Structure of Cascade Feed-Forward Back Propagation

4.2.2 Transfer Function

Table 4-2 shows a list of transfer (activation) functions in MATLAB's Toolbox. The mostly used transfer functions in this study were Hyperbolic Tangent Sigmoid (tansig) and Linear (purelin). The 'tansig' function is a non-linear function that is often used for pattern recognition problems. It calculates results and compresses them between -1 and 1 by Equation (6). The 'tansig' function performed the best set of operations after a series of training tests, this is why 'tansig' was selected as the primary transfer function in the network.

$$tansig(n) = \frac{2}{1+e^{-2n}} - 1 \tag{6}$$

The linear transfer function 'purelin' can be applied in function fitting problems, and typically is applied to the last layer and the output layer of both ANN(s). The output of this function is the multiplication of the function by a constant (Hagan, Demuth, & Beale, 2009).

Table 4.2 Transfer Functions (Hagan, Demuth, & Beale, 2009)

Name	Input/Output Relation	MATLAB Function
Hard Limit	$a = 0 \quad n < 0$ $a = 1 \quad n \geq 0$	hardlim
Symmetrical Hard Limit	$a = -1 \quad n < 0$ $a = +1 \quad n \geq 0$	hardlims
Linear	$a = n$	purelin
Saturating Linear	$a = 0 \quad n < 0$ $a = n \quad 0 \leq n \leq 1$ $a = 1 \quad n > 1$	satlin
Symmetric Saturating Linear	$a = -1 \quad n < -1$ $a = n \quad -1 \leq n \leq 1$ $a = 1 \quad n > 1$	satlins
Log-Sigmoid	$a = \frac{1}{1+e^{-n}}$	logsig
Hyperbolic Tangent Sigmoid	$a = \frac{e^{n} - e^{-n}}{e^{n} + e^{-n}}$	tansig
Positive Linear	$a = 0 \quad n < 0$ $a = n \quad 0 \leq n$	poslin
Competitive	$a = 1$ neuron with max n $a = 0$ all other neurons	compet

4.2.3 Training Function

The training algorithm is used to analyze and update the weighting factor for each neuron in the hidden layers. Table 4-3 shows a list of training functions in MATLAB's Toolbox. The scaled conjugate gradient function (trainscg) was selected due to its faster convergence rate among all other back propagation training functions and due to its insensitivity on user input (Meiller, 1993). The scaled conjugate gradient function was first developed by Martin Moller in 1990 to reduce the time for calculation and improve the efficiency of the entire process. Although it does more iterations than other conjugate gradient algorithms, it still dramatically decreases the total time on the computations by applying model trust region and conjugate approaches, and avoiding line search, in addition, it also has more storage requirements than other basic algorithms (Demuth & Beale, 1992).

Table 4.3 Training Functions (Vacic, 2005)

Function Name	Algorithm
trainb	Batch training with weight & bias learning rules
trainbfg	BFGS quasi-Newton backpropagation
trainbr	Bayesian regularization
trainc	Cyclical order incremental training w/learning functions
traincgb	Powell -Beale conjugate gradient backpropagation
traincgf	Fletcher-Powell conjugate gradient backpropagation
traincgp	Polak-Ribiere conjugate gradient backpropagation
traingd	Gradient descent backpropagation
traingdm	Gradient descent with momentum backpropagation
traingda	Gradient descent with adaptive lr backpropagation
traingdx	Gradient descent w/momentum & adaptive lr backpropagation
trainlm	Levenberg-Marquardt backpropagation
trainoss	One step secant backpropagation
trainr	Random order incremental training w/learning functions
trainrp	Resilient Backpropagation (Rprop)
trains	Sequential order incremental training w/learning functions
trainscg	Scaled conjugate gradient backpropagation

4.2.4 Learning Function

Table 4.4 shows a list of learning functions that are commonly used for neural networks in MATLAB Toolbox. The applied learning function for backpropagation networks in this study is the gradient descent with momentum weight and bias learning function ('learngdm'). Refer to Equation (7), this learning function calculates the change of weight based on input and error of the neuron, the learning rate, the weight, and momentum constant (Demuth & Beale, 1992).

$$dW = mc * dWprev + (1 - mc) * lr * gW \tag{7}$$

Where MC is momentum constant, and LR stands for learning rate.

Table 4.4 Learning Functions (Demuth & Beale, 1992)

Function Name	Algorithm
Learncon	Conscience bias learning function
Learngd	Gradient descent weight/bias learning function
Learngdm	Grad. Descent w/momentum weight/bias learning function
Learnp	Perceptron weight/bias learning function
Learnpn	Normalized perceptron weight/bias learning function
Learnwh	Widrow-Hoff weight/bias learning rule

4.2.5 Performance Function

Table 4.5 shows a list of performance functions that are commonly used for neural networks in MATLAB Toolbox. The default performance function for feedforward networks is a mean square error ('mse'), which is the average squared error between the network outputs and the target outputs. Equation (8) explains the algorithm of the 'mse' function with 'a' represents network outputs and target outputs represented by 't'. In this study, the mean sum of squares of the network errors ('msereg'), which is the typical performance function modified based on the default performance function, was applied to both the forward ANN and the inverse ANN. Besides the mean squared error, it also measures network performance on the mean squared weights and biases. Equation (9) below explains the algorithm of 'msereg' with 'γ' as the performance ratio. The application of this performance gave the best converge rate because it led to smaller weights and biases in the network that force the network to perform more smoothly and reduce the occurrence of overfitting issues.

$$F = mse = \frac{1}{N}\sum_{i=1}^{N}(e_i)^2 = \frac{1}{N}\sum_{i=1}^{N}(t_i - a_i)^2 \tag{8}$$

$$mse = \gamma mse + (1-\gamma)\frac{1}{n}\sum_{j-1}^{n} w_j^2 \tag{9}$$

Table 4.5 Performance Functions (Demuth & Beale, 1992)

Function Name	Algorithm
mae	Mean absolute error-performance function
mse	Mean squared error performance
msereg	Mean squared error w/reg performance function
sse	Sum squared error performance function

4.3 Forward Neural Network

The goal of construction of a forward neural network is to predict the rate transient data rapidly with time by knowing reservoir properties and well design parameters, as shown in Figure 4.3. As mentioned previously, the input reservoir parameters are reservoir pattern size, reservoir depth, reservoir thickness, matrix porosity, matrix permeability, fracture porosity, fracture permeability, fracture spacing, fracture compressibility, matrix compressibility, reservoir temperature, reservoir pressure, and bottomhole pressure for the forward neural network. The input well design parameters were depths and lengths of two lateral wells. The output targets were 10-year field production rates at four months interval. Thus, there were 17 input elements and 31 production targets applied in the forward neural network training process.

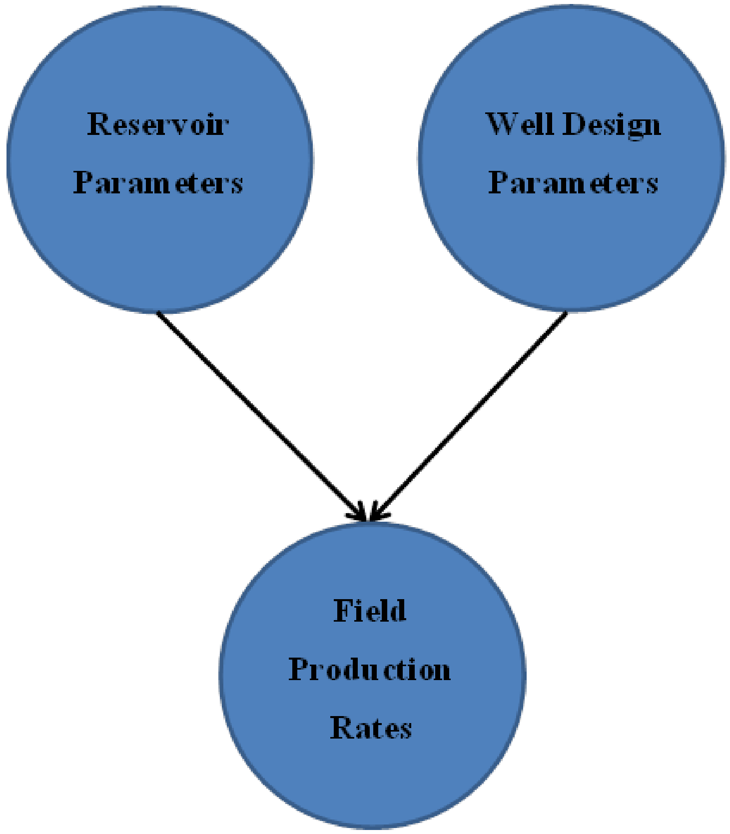


Figure 4.3 Forward Solution Diagram

After performing several tests on the different combinations of ANN components, the optimal combination of the forward artificial neural network was achieved and which consisted of three hidden layers with each layer containing 25, 25, and 20 neurons, respectively. The transfer function for each hidden layer is tan-sigmoid (tansig) and the learning function selected is the gradient descent type with momentum weight/bias function (learngdm). The error rule was based on the mean sum of squared error performance function (msereg). The total 1965 data sets were randomly divided by the 'dividerand' and 'divideind' functions into training sets, validation sets, and testing sets at the proportions of 70%, 15%, and 15%. Figure 4.4 below illustrates the final optimal structure for the forward artificial neural network. Table 4.6 illustrates sample tests on the structure combinations based on the regression performance.

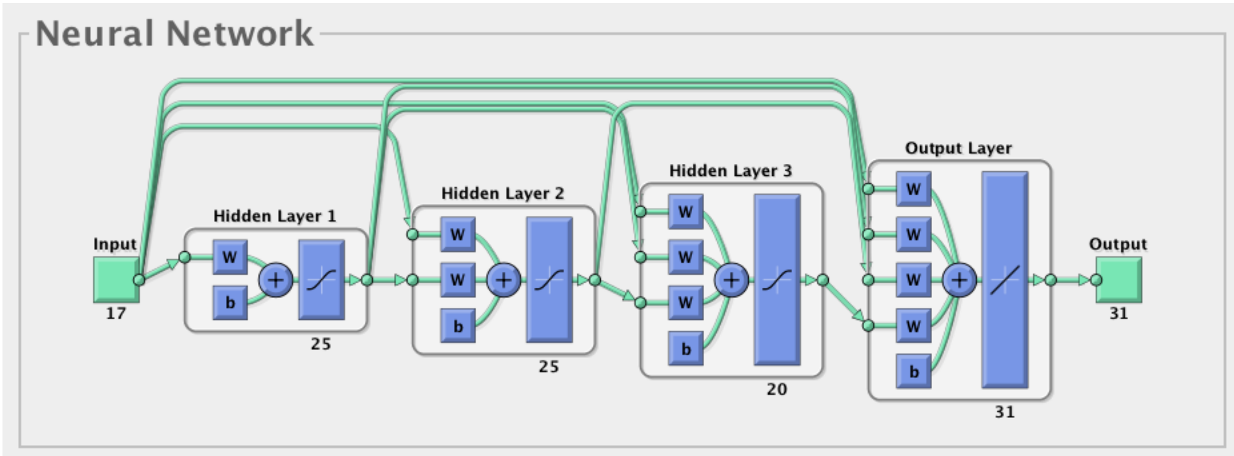


Figure 4.4 Forward ANN Structure in Training

Table 4.6 Sample Tests of ANN Structure

		Regression Parameters			
Number of Hidden Layers	Neuron Combinations	Training	Validation	Testing	All
2	20,20	0.99626	0.99521	0.99586	0.99605
2	25,25	0.9981	0.99712	0.99626	0.99767
2	30,30	0.9977	0.99571	0.99698	0.9973
2	35,35	0.99734	0.99631	0.99677	0.99707
3	20,20,20	0.99714	0.99496	0.99548	0.99665
3	25,25,25	0.99741	0.99615	0.99546	0.99698
3	25,25,20	0.99917	0.99824	0.99847	0.99894
3	30,30,30	0.99649	0.99592	0.99623	0.99636
3	35,35,35	0.99746	0.99616	0.99503	0.99693
4	20,20,20,20	0.99748	0.99598	0.99716	0.9972
4	25,25,25,25	0.99726	0.99504	0.99432	0.9965
4	30,30,30,30	0.9968	0.99536	0.99515	0.99631
4	35,35,35,35	0.99602	0.99367	0.99399	0.99538

4.4 Inverse Neural Network

The goal of construction of an inverse neural network was to predict the desired reservoir characterizations rapidly by knowing the other reservoir properties, well design parameters, and field production rate, as shown in Figure 4.5. As mentioned earlier, for the inverse neural network, the input reservoir parameters are reservoir pattern size, reservoir depth, fracture compressibility, matrix compressibility, reservoir temperature, reservoir pressure, and bottomhole pressure. The input well design parameters are depths and lengths of two lateral wells. The input ten field production rates were selected from first 3-year production rates with an interval of 120 days. The reason for choosing these field production rates is because of the fast decline happens at an early stage of prediction so that the neural network can recognize the declining pattern to give better predictions on the desired parameters. In addition, the desired parameters are preferred to be estimated at the beginning of the production. The output reservoir characterization targets are reservoir thickness, matrix porosity, matrix permeability, fracture porosity, fracture permeability, and fracture spacing. The training results were improved by adding last 20 field production rates at 120-day interval as target output. These production rates are just used for training purpose to improve the accuracy of the desired parameters and not applied to any further process. Thus, there were 21 input elements and 26 output targets but only 6 desired ones were evaluated and analyzed. Table 4.7 shows the significant improvement on the training regression parameters by adding the last 20 field production rates as target elements.

Table 4.7 Effects on Adding Production Profile as Target

	Number of Hidden Layers	Neuron Combinations	Regression Parameters			
			Training	Validation	Testing	All
Case without 20 Production Rate as Target	4	30,30,30,25	0.73953	0.67312	0.68787	0.72194
Case with 20 Production Rate as Target	4	30,30,30,25	0.91865	0.89955	0.90029	0.913

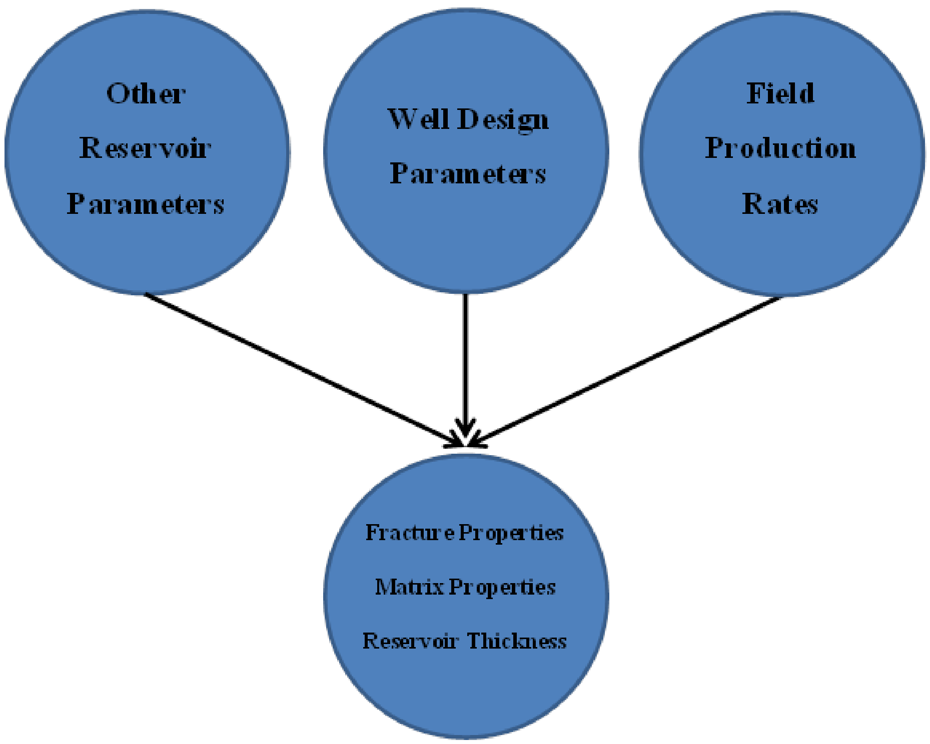


Figure 4.5 Inverse Solution Diagram

After several tests carried out on the different combinations of ANN components, an optimal combination of the inverse artificial neural network consisting of four hidden layers, which contains 30, 30, 30, and 25 neurons, respectively, was achieved. The transfer function for each hidden layer was tan-sigmoid (tansig) and the learning function selected was the gradient descent type with momentum weight/bias function (learngdm). The error rule was based on the mean sum of squared error performance function (msereg). The total 1965 data sets were randomly divided by the 'dividerand' and 'divideind' functions into training sets, validation sets, and testing sets at the proportions of 70%, 15%, and 15% respectively. Figure 4.6 below illustrates the final optimal structure for the inverse artificial neural network and Table 4.8 illustrates sample tests on the structure combinations based on the regression performance.

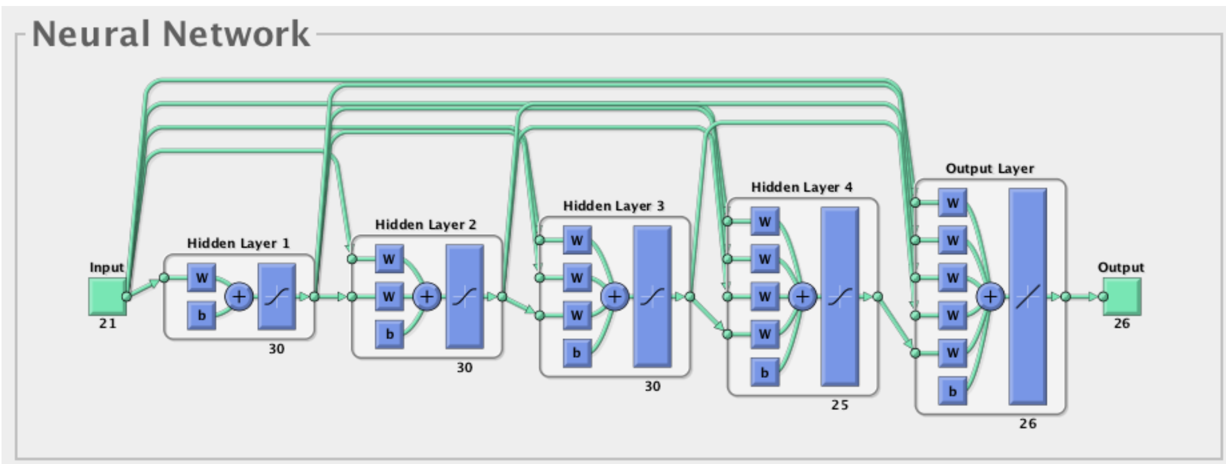


Figure 4.6 Inverse ANN Structure in Training

Table 4.8 Sample Tests of ANN Structure

		Regression Parameters			
Number of Hidden Layers	Neuron Combinations	Training	Validation	Testing	All
2	20,20	0.88635	0.87946	0.86452	0.88228
2	25,25	0.89898	0.86752	0.86636	0.88988
2	30,30	0.88738	0.87042	0.86027	0.88112
2	35,35	0.88462	0.86371	0.87219	0.87963
3	20,20,20	0.89502	0.89022	0.87783	0.89189
3	25,25,25	0.89826	0.88534	0.87799	0.89344
3	30,30,30	0.89873	0.8747	0.88164	0.89259
3	35,35,35	0.9028	0.87234	0.86795	0.89326
4	20,20,20,20	0.90281	0.87543	0.8709	0.89418
4	25,25,25,25	0.8982	0.87277	0.87089	0.89085
4	30,30,30,25	0.91865	0.89955	0.90029	0.913
4	30,30,30,30	0.91254	0.88058	0.89496	0.90537
4	35,35,35,35	0.91022	0.88709	0.88153	0.90251

Chapter 5: Results and Discussion

In this chapter, the results from both forward ANN and inverse ANN models will be presented, analyzed, and discussed. For forward ANN, the performance of the training process and the test error distribution will be considered. The predicted outputs will be analyzed along with the target outputs by scattered data method. There will be a total of five predicted curves to be discussed, three of them are randomly picked from each error distribution group, and the other two are the cases with maximum and minimum errors. For inverse ANN, the performance of the training process will also be presented. In addition, the error distribution for each desired reservoir prediction parameter will be analyzed and discussed.

5.1 Forward ANN Results

The performance of forward ANN training is shown below in Figure 5.1. The best performance is achieved at epoch number 6075, the iteration at which the validation performance reached a minimum. The early stopping technique is the default method for improving generalization and is automatically provided for all of the supervised network creation functions. It divides the data into training, validation, and testing sets. The validation data is monitored during the training process, and the error of the validation is expected to decrease when the error of training sets decreases. The network begins to overlearn the data when the validation error begins to increase, and early stopping technique should be applied to training process to prevent this phenomenon.

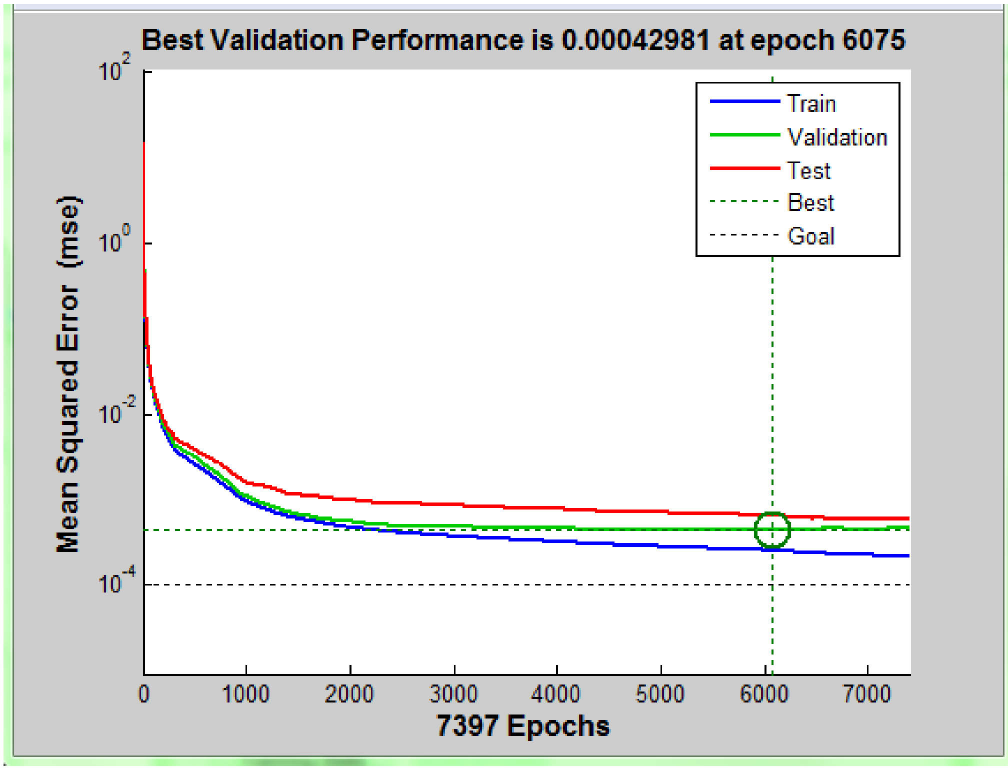


Figure 5.1 Performance of Forward ANN Training

The regression of forward ANN training set is shown below in Figure 5.2. The correlation between the outputs and the targets is very high due to all the R parameters are in the proximity of 1.

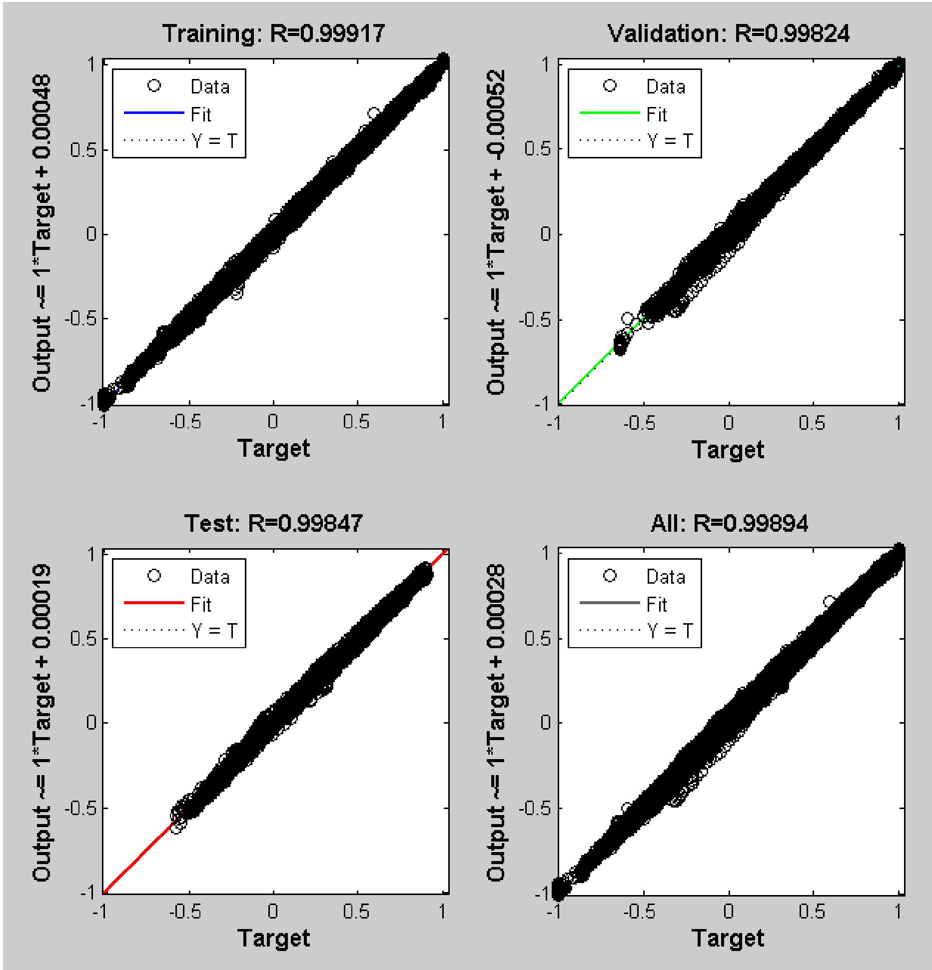


Figure 5.2 Regression Plots for Forward ANN Training

The relative error calculations and their percentages are displayed in Figure 5.3 below. It is shown that 87.99% of the cases have relative errors less than 0.5%. It is also shown that 10.79% of the cases have a relative errors between 0.5% and 1%, and only 1.21% of the cases have relative errors larger than 1%.

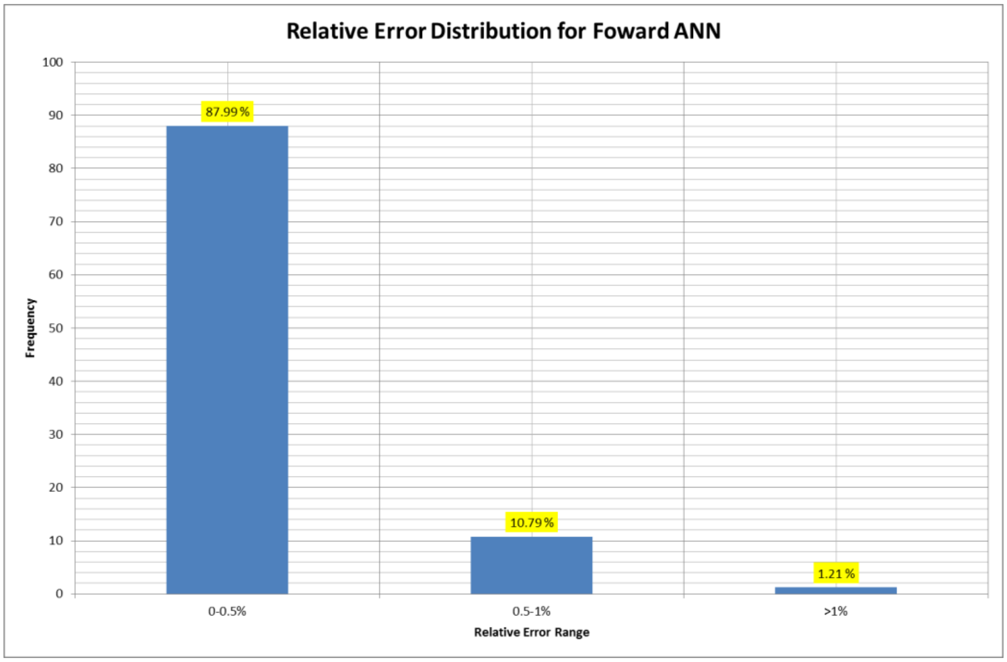


Figure 5.3 Relative Error Distributions for Forward ANN

Figure 5.4 below shows a production profile with a relative error of 0.346%. This case is randomly selected from the group with relative error less than 0.5%. It can be seen that the trained forward network shows an impressive performance in matching the decline curve with respect to the simulation results from the numerical model. The predicted curve has larger divergence before 400 days of production.

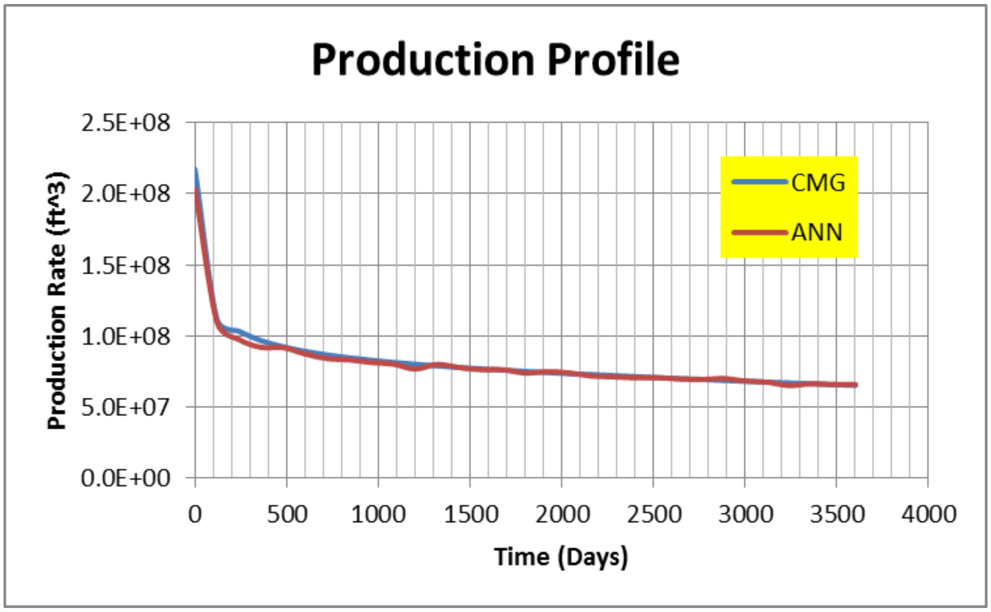


Figure 5.4 Random Production Profile with Low Error

Figure 5.5 below shows a production profile with a relative error of 0.5795%. This case is randomly selected from the group with a relative error margin between 0.5% and 1%. Compared to the ones in Figure 5.4, this case has a less accurate prediction. The entire production data are predicted less than the target values, and thus the predicted curve is slightly lower than the targeted curve. In addition, for this case, the significant divergence happens before 1,500 days of production.

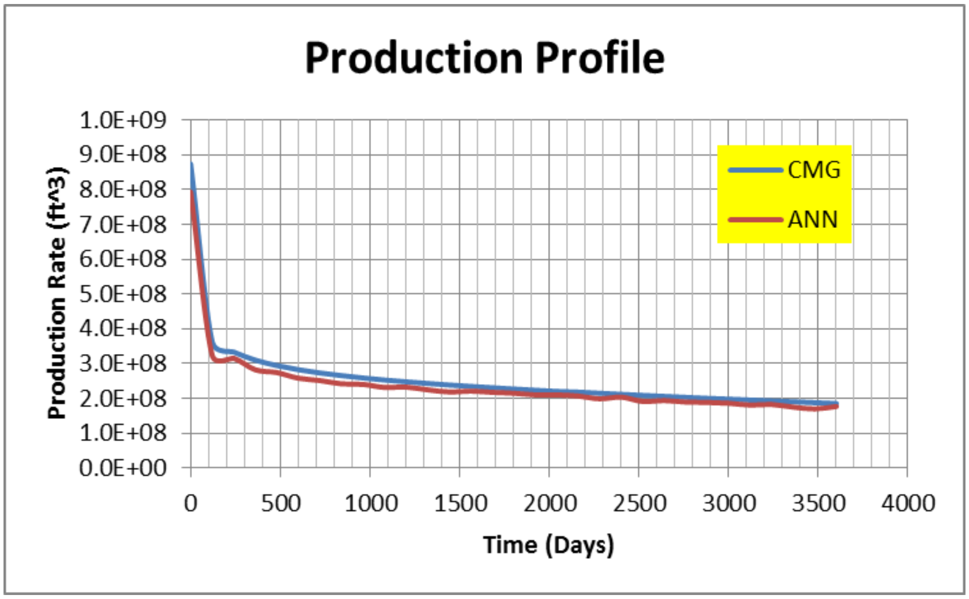


Figure 5.5 Random Production Profile with Medium Error

Figure 5.6 below shows a production profile with a relative error of 1.3645%. This case is randomly selected from the group with a relative error greater than 1%. As the plot shows, the significant divergence happens before 1,000 days of production.

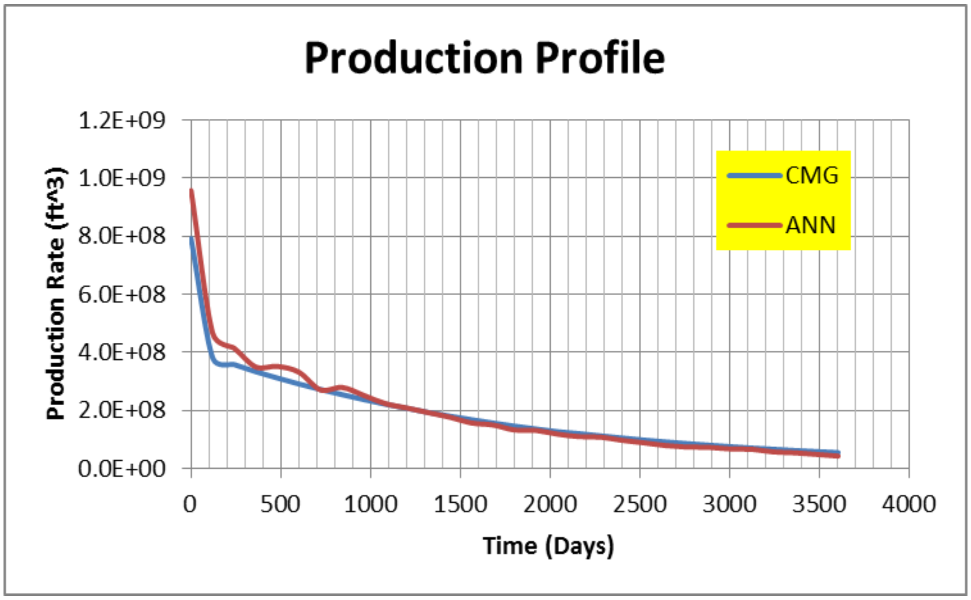


Figure 5.6 Random Production Profile with High Error

Figure 5.7 below shows the case with a maximum error of 2.397%. Generally, the entire predicted curve is above the simulated CMG production curve, and the significant divergence happens before 1,800 days of production. Although it is the case with a maximum error of this trained network, 2.397% is still considered as a comparatively low error rate for the artificial neural network topology developed.

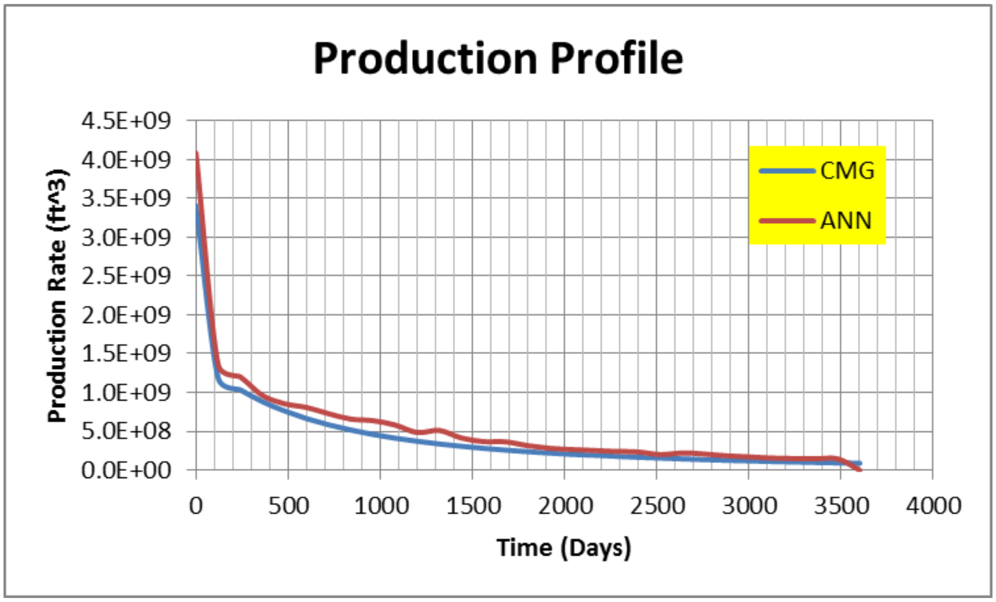


Figure 5.7 Production Profile with Maximum Error

Figure 5.8 below displays the case with the minimum error of 0.0000708%. The predicted production curve almost overlaps the simulated curve, and it only has minute divergences between 100 days and 400 days of production.

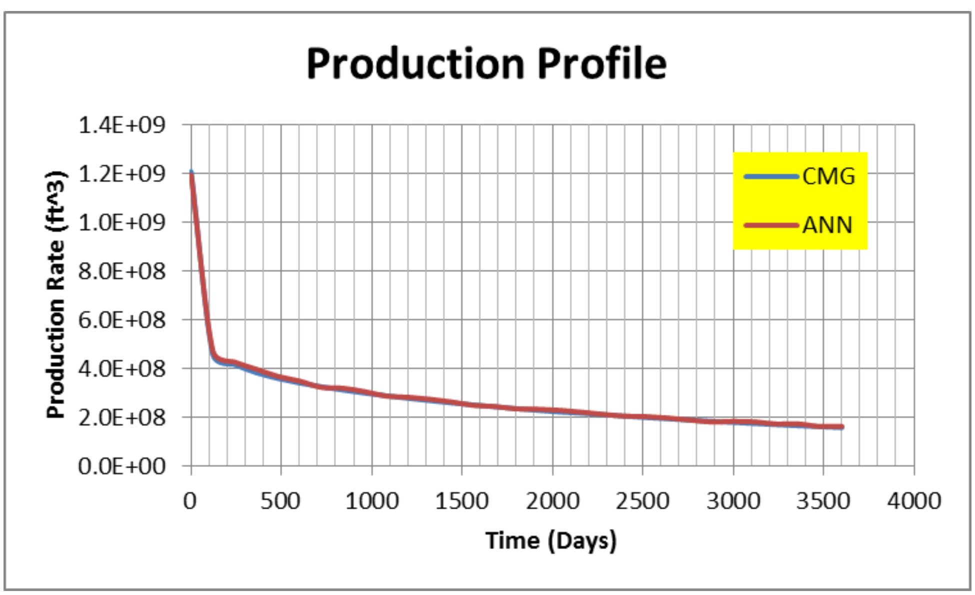


Figure 5.8 Production Profile with Minimum Error

Although, most cases that display divergences happen at the early stages of the production, the entire curves are considered as precise predictions because they are either very close to or overlapping with the simulated results from the numerical model with insignificant divergences. In addition, the high degree of correlation between targets and outputs from regression plots also indicates the high performance at the training stage and its superior performance in predicting field production rates during the 10-year production period.

5.2 Inverse ANN Results

The performance of inverse ANN training is shown below in Figure 5.9. The best performance is achieved at epoch number 1170, which indicates the iteration at which the validation performance reached a minimum, which was achieved by early stopping method.

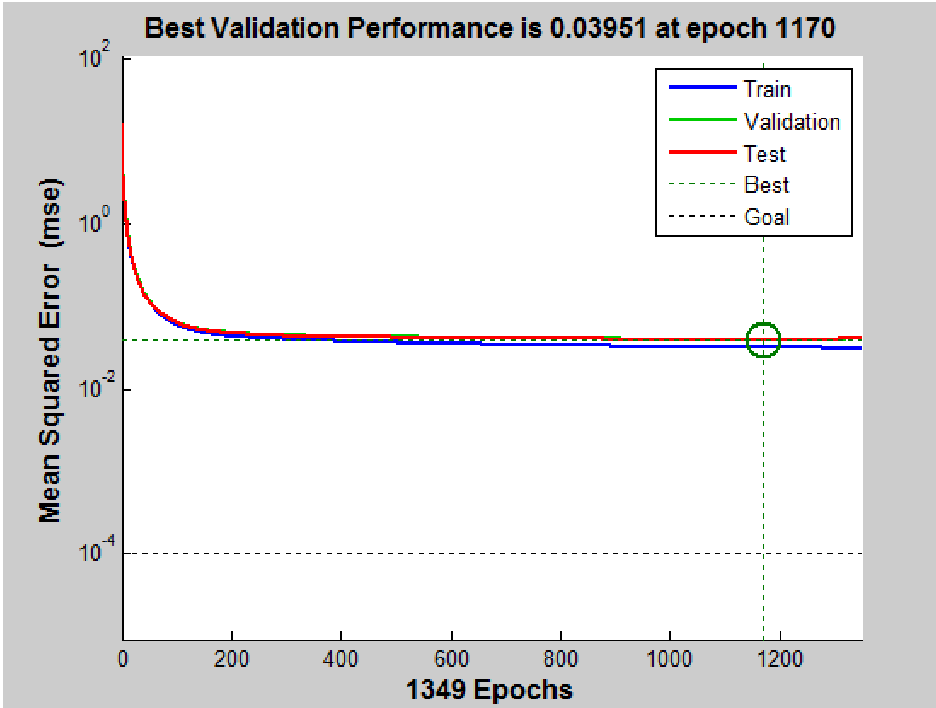


Figure 5.9 Performance of Inverse ANN Training

5.2.2 Sensitivity Check

In order to determine the cut offs of the error margin of each desired reservoir parameters, the sensitivity each of parameter is simulated by the CMG software. The methodology is to changing one desired parameter at each time and simulated by numerical model to compare the production profile with the original decline curve. The original decline curve is simulated from the numerical model with mean values of all the parameters in Table 4.1.

From Figure 5.10-5.15, it can be noticed that the hydrocarbon production is influenced insignificantly by fracture permeability with less than 10% of variation. It can be concluded that fracture porosity, matrix permeability, and matrix porosity has negligible impact on production profiles with variations less than 10%. It can also be observed that variation of thickness less than 10% has only small impact on deviation of production rate from the original data. Fracture spacing has negligible effects on production rate with variation as large as 120%.

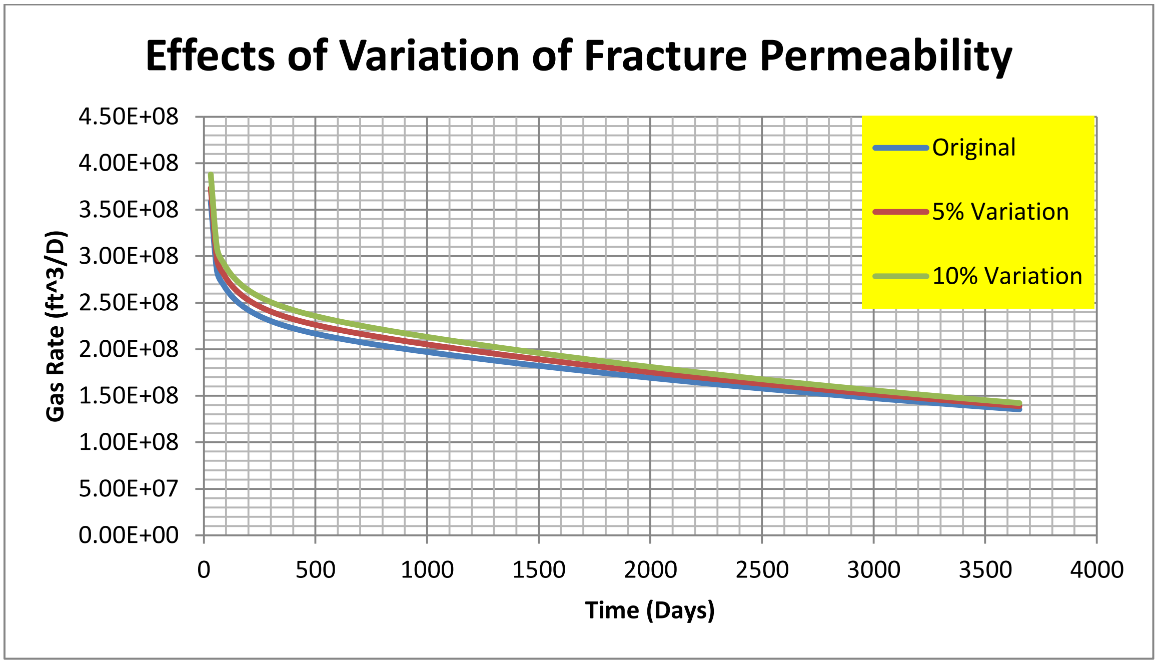


Figure 5.10 Effects of Variation of Fracture Permeability

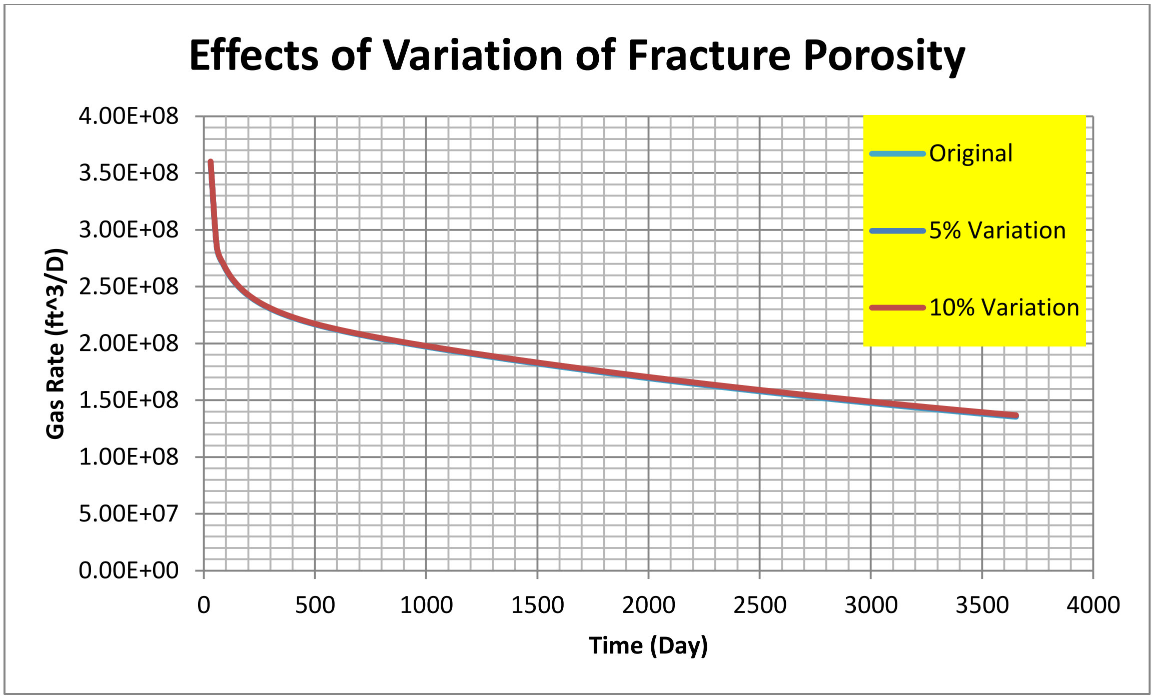


Figure 5.11 Effects of Variation of Fracture Porosity

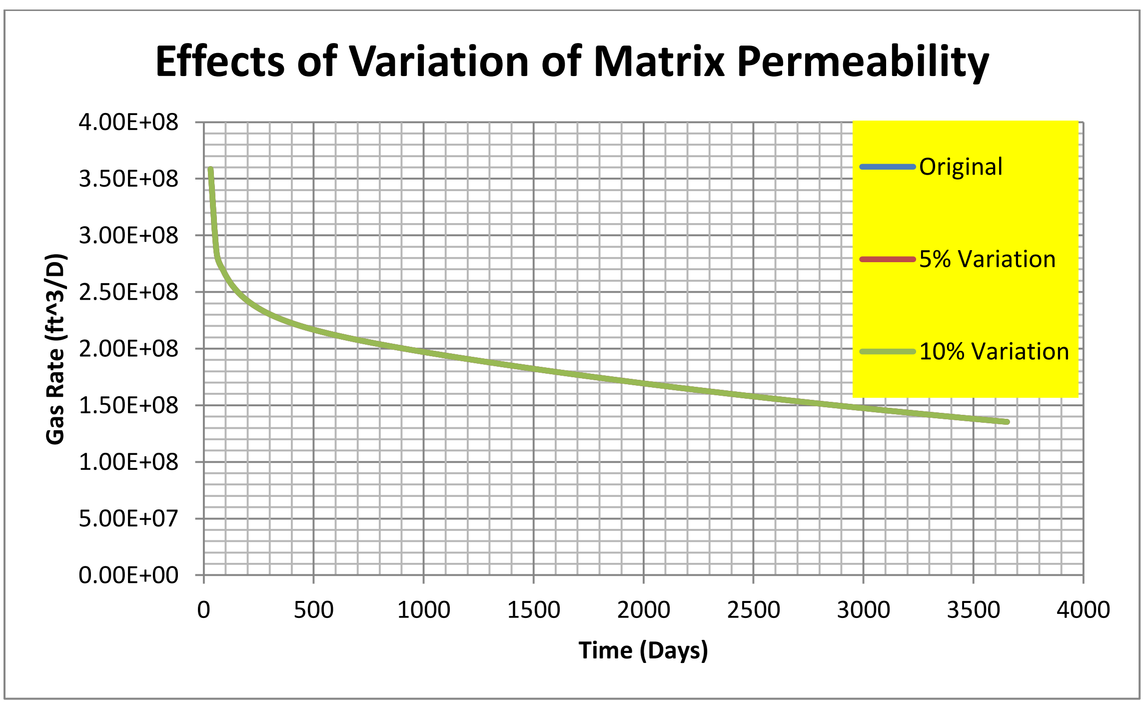


Figure 5.12 Effects of Variation of Matrix Permeability

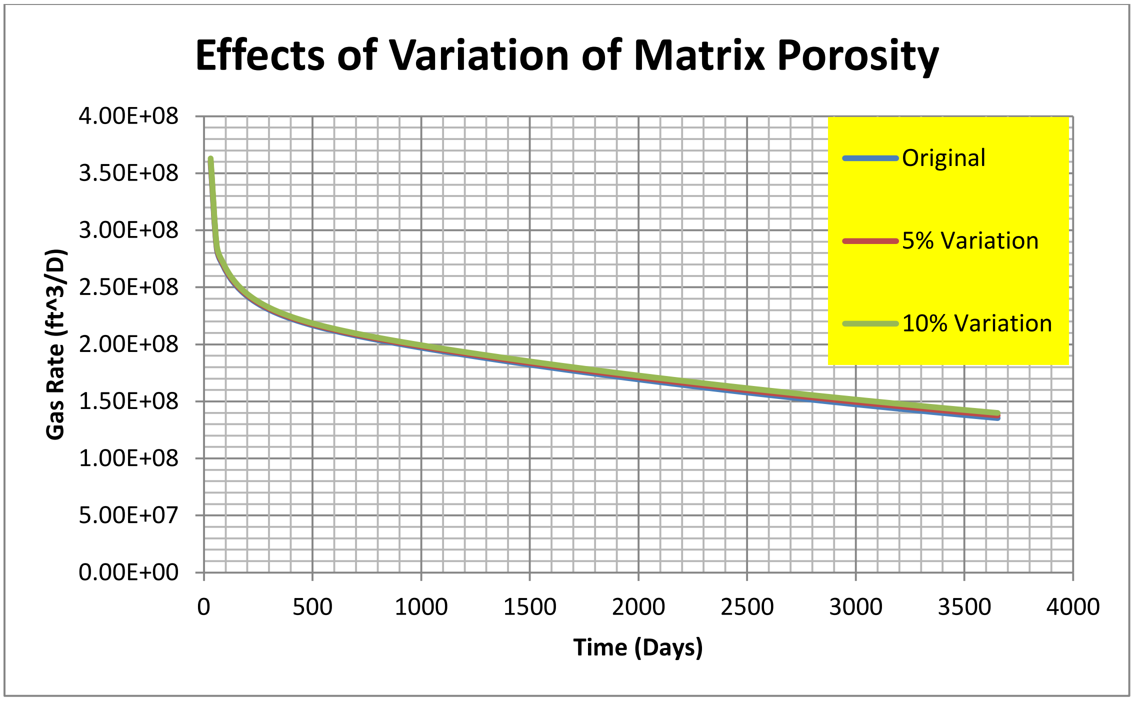


Figure 5.13 Effects of Variation of Matrix Porosity

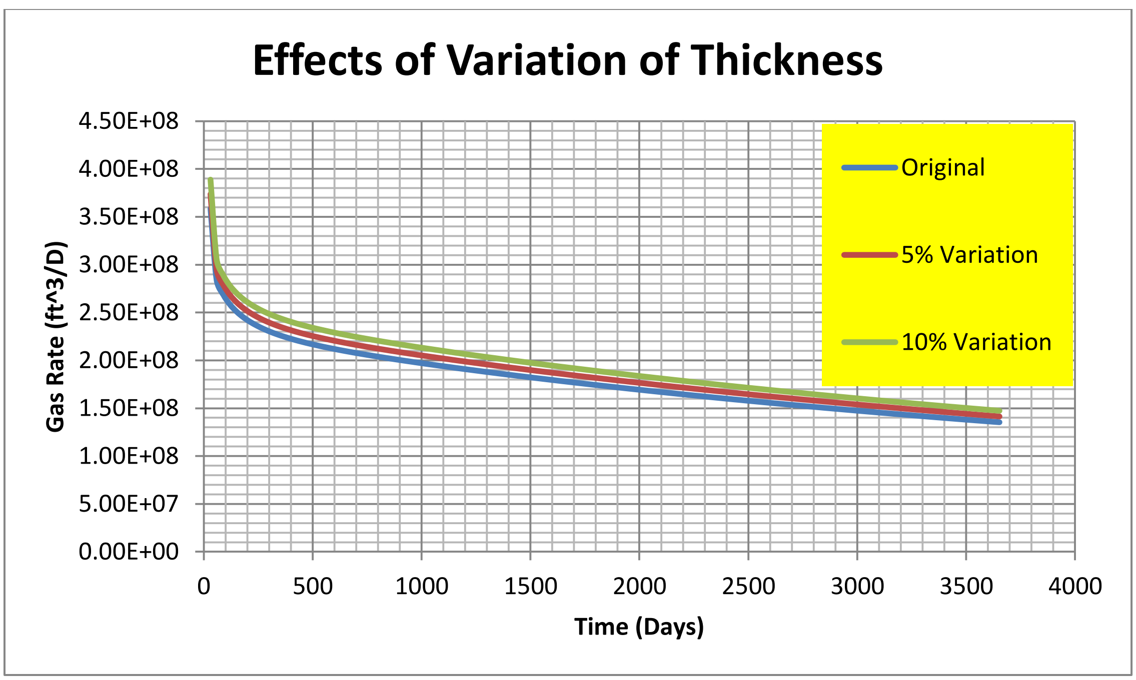


Figure 5.14 Effects of Variation of Thickness

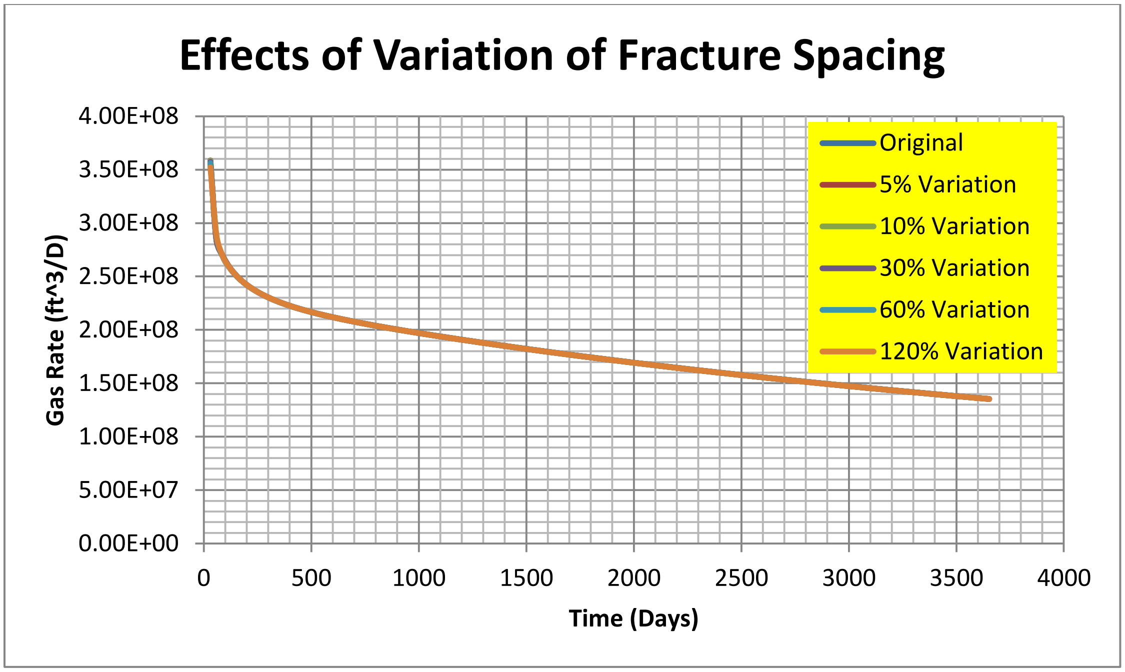


Figure 5.15 Effects of Variation of Fracture Spacing

5.2.3 Error Distributions

The error distribution for normalized fracture permeability, fracture porosity, fracture spacing, thickness, matrix permeability, and matrix porosity are shown in Figure 5.16, Figure 5.17, and Table 5.1. The minimum and maximum errors for fracture permeability are 0.015% and 12.51%, respectively. In addition, 85.08% of the data shows error margins below 5% and only 1.02% of total test data has error margins larger than 10%. The minimum and maximum errors for fracture porosity are 0.0028% and 6.49%, respectively, and 91.19% of the data displays error margins below 5%. The minimum and maximum errors for fracture spacing are 0.062% and 671.70%, respectively. It is also noted that 22.71% of the data has an error less than 5%, and 53.56% of total test data has error more than 10%. The minimum and maximum errors for formation thickness are 0.020% and 10.28%, respectively. In addition, 87.80% of the data has error below

5% and only 0.34% of total test data has error ranges greater than 10%. The minimum and maximum errors for matrix permeability are 0.051% and 12.99%, respectively. In addition, 67.80% of the results have error margins below 5% and only 0.68% of total test data has error larger than 10%. The minimum and maximum errors for matrix porosity are 0.0016% and 20.72%, respectively. Moreover, 66.78% of the results have error margins below 5%, and only 6.44% of total test data has error greater than 10%.

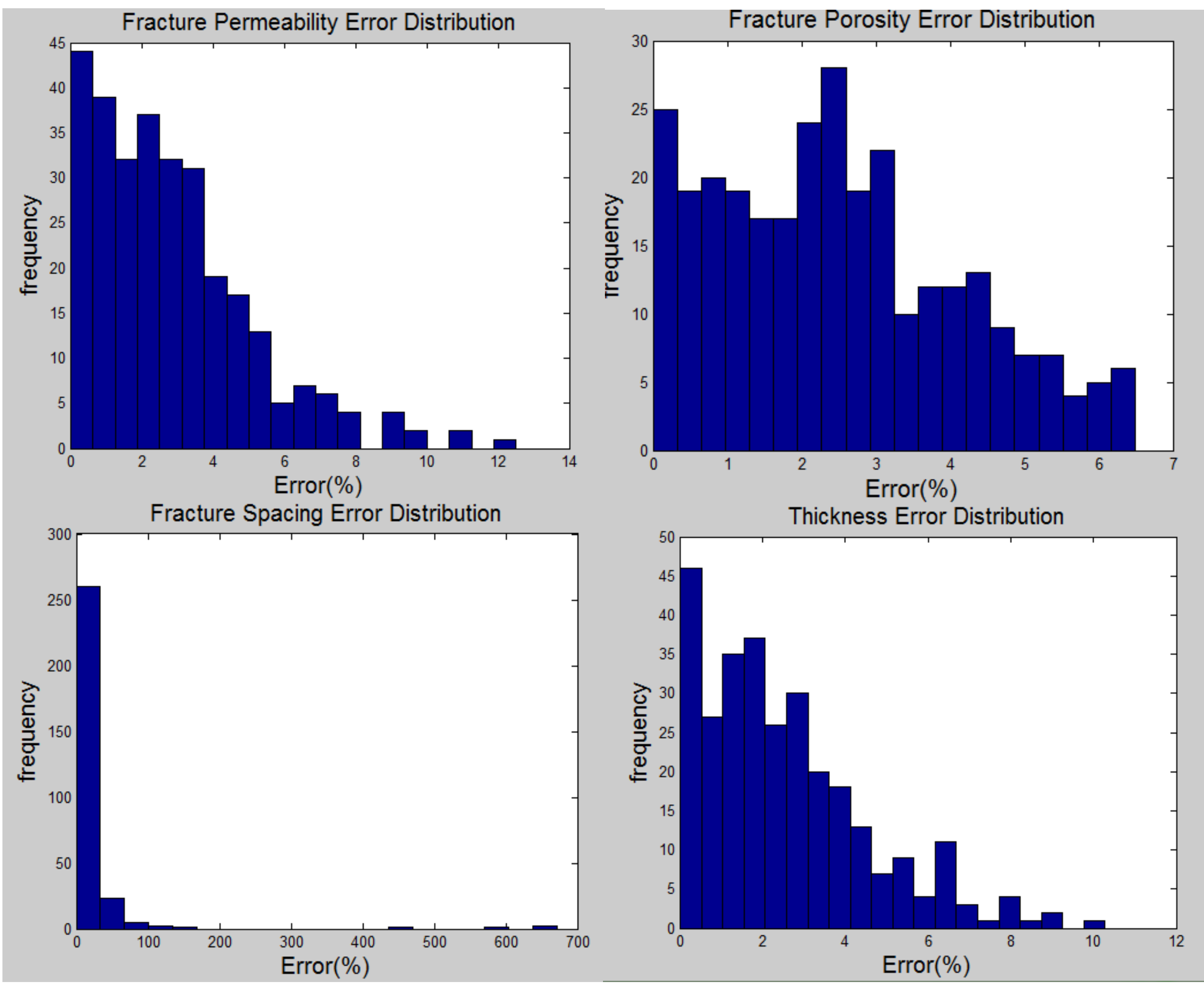


Figure 5.16 Error Distributions for Fracture Characterizations and Thickness

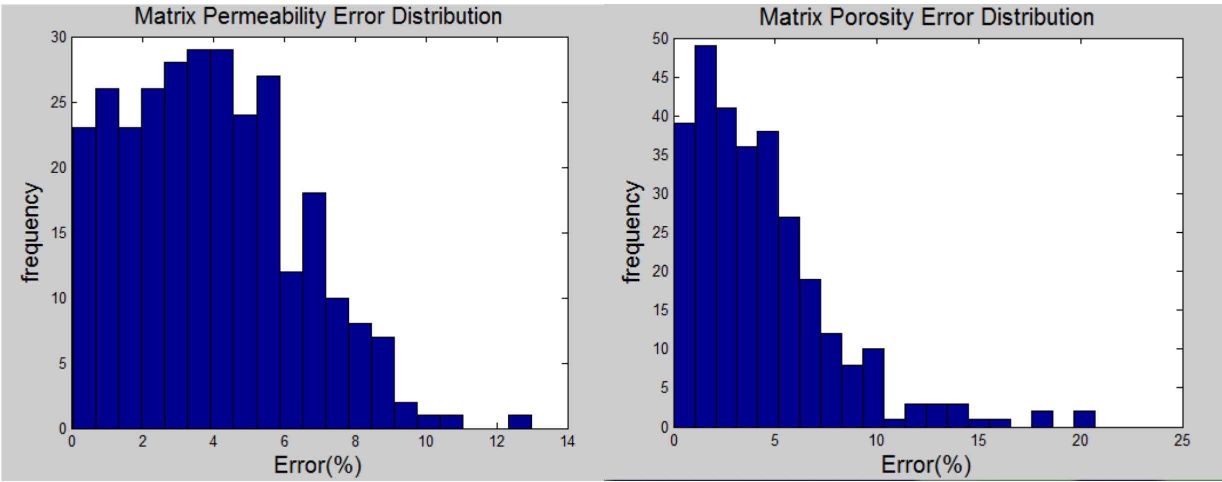


Figure 5.17 Error Distributions for Matrix Characterizations

Table 5.1 Summary of Error Distribution for Inverse ANN

	Error Distribution											
Error Range (%)	**Fracture Permeability**		**Fracture Porosity**		**Fracture Spacing**		**Formation Thickness**		**Matrix Permeability**		**Matrix Porosity**	
0-5	**251**	**85.08%**	**269**	**91.19%**	**67**	**22.71%**	**259**	**87.80%**	**200**	**67.80%**	**197**	**66.78%**
5-10	**41**	**13.90%**	**26**	**8.81%**	**70**	**23.73%**	**35**	**11.86%**	**93**	**31.53%**	**79**	**26.78%**
>10	**3**	**1.02%**	**0**	**0.00%**	**158**	**53.56%**	**1**	**0.34%**	**2**	**0.68%**	**19**	**6.44%**

As shown in Figures 5.18 through 5.23, 20 data sets were randomly selected from the inverse network training testing results. The predicted reservoir properties are highly matching with the reservoir model data. It is noted that thickness deviations from 5.02 ft to 54.79 ft, fracture spacing deviates from 0.23 ft to 82.61 ft, fracture permeability deviates from 0.0003 mD to 0.14 mD, matrix porosity deviates from 0.12% to 10.65%, matrix permeability deviates from 0.003 mD to 0.028 mD, and fracture porosity deviates from 0.06% to 1.49%. Based on these results, it can be considered that the inverse ANN successfully and precisely predicts the fracture properties, matrix properties, and reservoir thickness with a three-year production profile and other reservoir properties. Thus, it can be concluded that the trained inverse artificial neural network is a reliable tool for predicting these parameters.

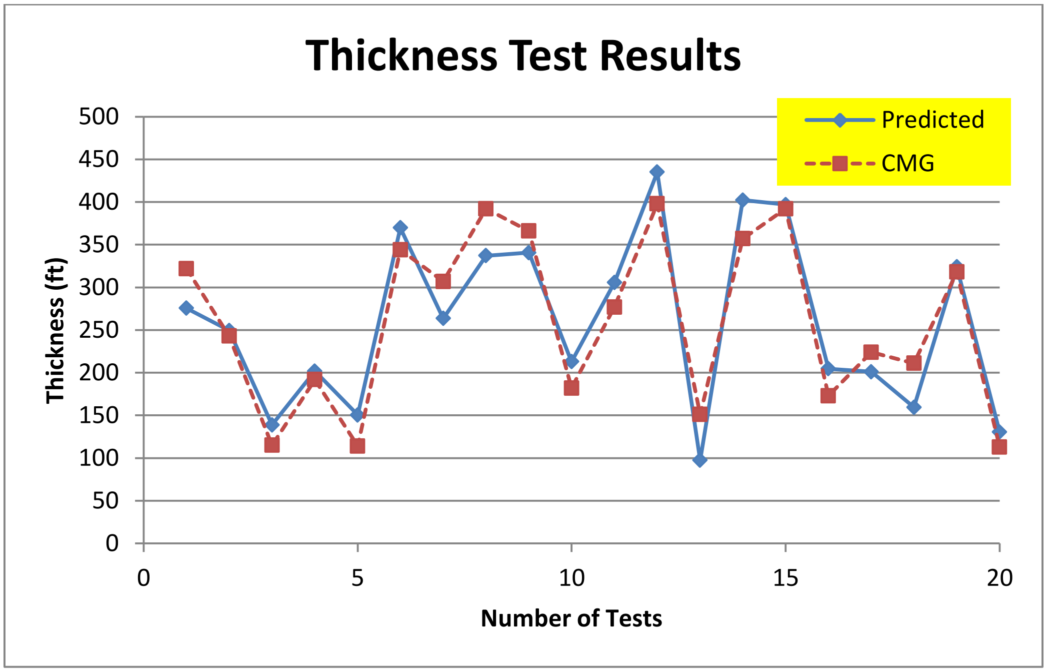


Figure 5.18 Sample Thickness Testing Results

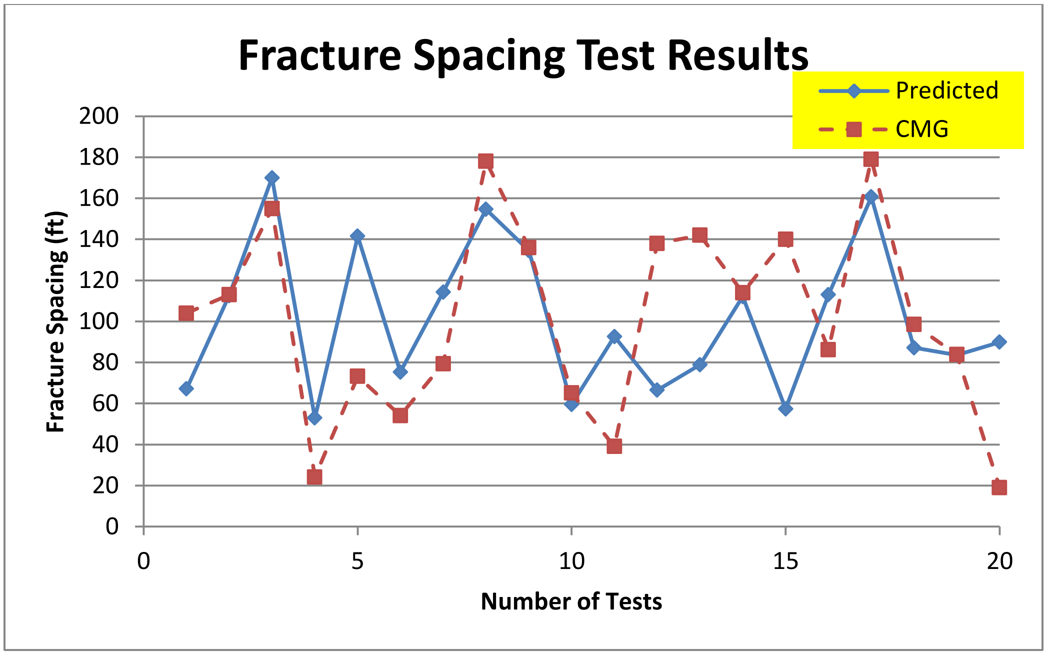


Figure 5.19 Sample Fracture Spacing Testing Results

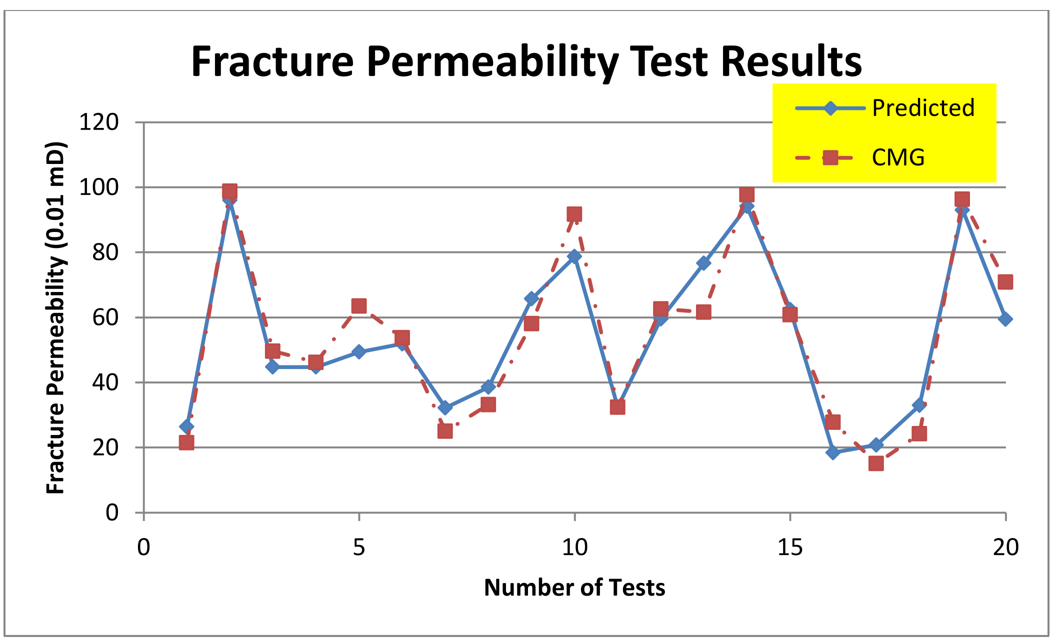


Figure 5.20 Sample Fracture Permeability Testing Results

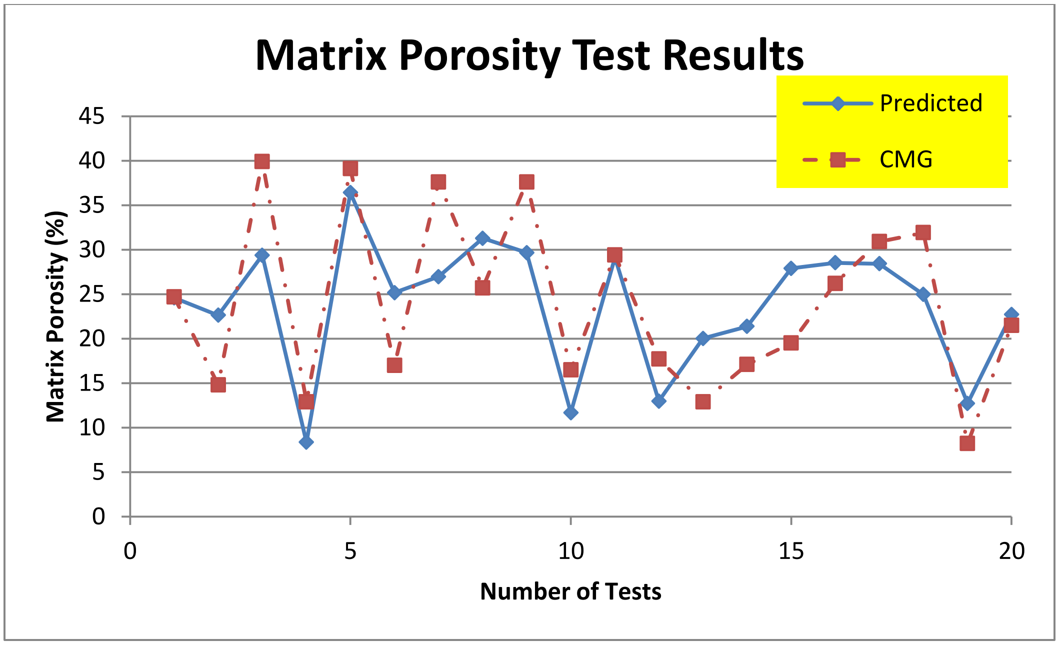


Figure 5.21 Sample Matrix Porosity Testing Results

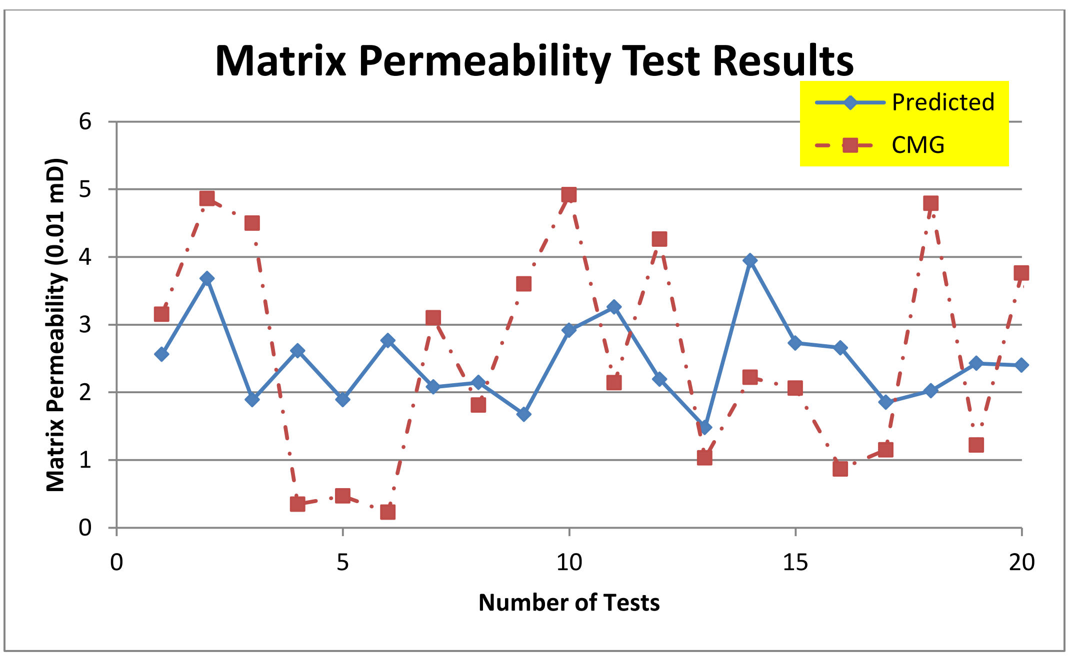


Figure 5.22 Sample Matrix Permeability Testing Results

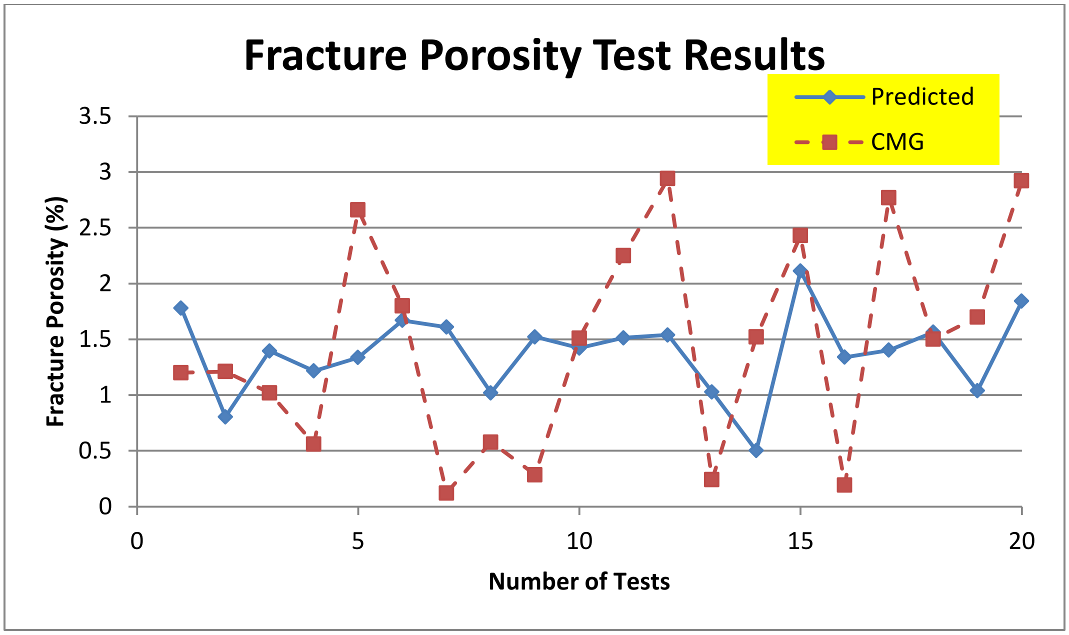


Figure 5.23 Sample Fracture Porosity Testing Results

Chapter 6: Development of a Graphic User Interface

Once the forward and inverse artificial neural networks are established, it is necessary to develop a Graphical User Interface (GUI) to allow the user to generate desired data predictions in a more efficient and straightforward way without recourse to further coding.

6.1 Graphical User Interface for Forward ANN

The GUI for forward artificial neural network prediction is shown in Figure 6.1. It has mainly five dialog boxes: Reservoir Parameters Input, Well Parameters Input, Production Output, Graphs, and Buttons. The Reservoir Parameters Input panel allows user to input area, depth, thickness, matrix porosity, fracture porosity, matrix permeability, fracture permeability, fracture space, matrix compressibility, fracture compressibility, reservoir temperature, reservoir pressure, and bottomhole pressure. The Well Parameters Input panel allows users to input lengths and depths of two lateral wells. The Production Output panel is used to display the detailed predicted daily production rate and the corresponding cumulative production data. The Graph panel is used to show the prediction results, visualize the daily production rate and to plot cumulative production file into two different graphs. The Buttons panel allows the user to predict data based on the inputs, to reset entire GUI, to use the help button to display the input ranges of all parameters, and to quit the GUI. In Figure 6.1, after a random data set was input, a 10-year daily production plot and a 10-year cumulative production plot were predicted and shown in the Graphs panel, the exact data is shown in the Production Output panel that shows approximately a production rate of 29-million cubic feet per day and a cumulative production of 2-billion cubic feet at the end of a 10-year period.

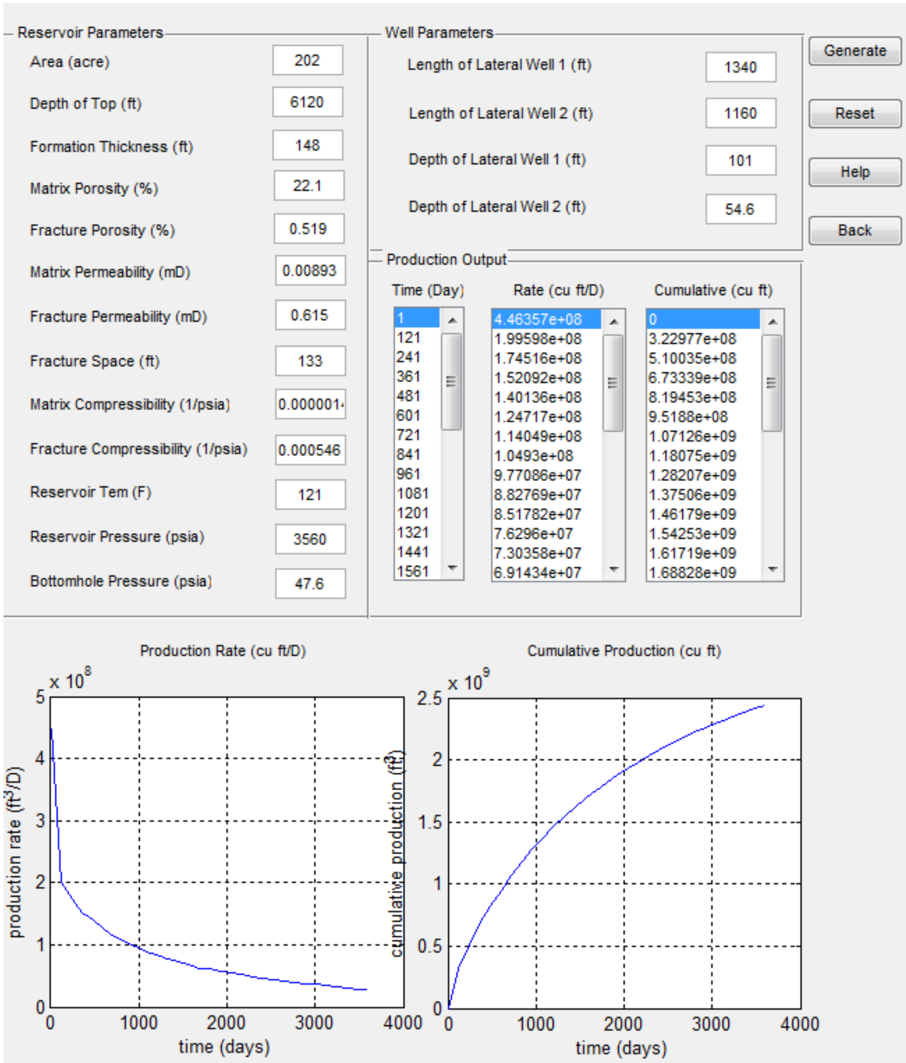


Figure 6.1 Graphic User Interface for Forward ANN

6.2 Graphical User Interface for Inverse ANN

The GUI for inverse artificial neural network prediction is shown in Figure 6.2. It is also divided into five sections: Reservoir Parameters Input, Well Parameters Input, Production Rate Input, Prediction Parameters, and Buttons. The Reservoir Parameters Input panel allows the user to input area, depth, matrix compressibility, fracture compressibility, reservoir temperature, reservoir pressure, and bottomhole pressure. The Well Parameters Input panel allows users to input lengths and depths of two lateral wells. The Production Input panel allows the user to input monthly production rate every four months for a period of three years. The Prediction Parameters panel is used to display the prediction results, which includes thickness, matrix porosity, fracture porosity, matrix permeability, fracture permeability, and fracture spacing. The Buttons panel allows the user to predict results based on the inputs, reset entire GUI, to use the help button to display the ranges of all parameters, and to quit the GUI. In Figure 6.2, after a random data set was input, the matrix porosity, fracture porosity, fracture spacing, matrix permeability, fracture permeability, and reservoir thickness were directly predicted and displayed on the bottom panel with corresponding units.

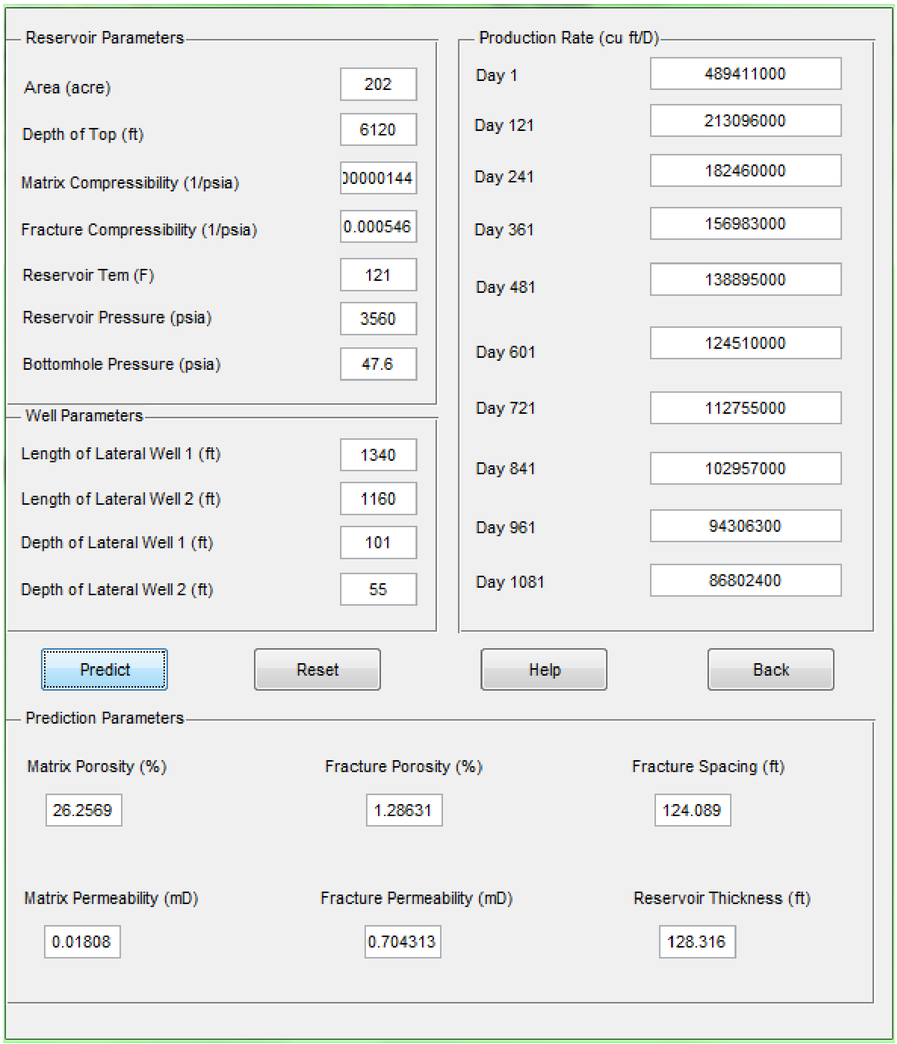


Figure 6.2 Graphic User Interface for Inverse ANN

Chapter 7: Summary and Conclusion

The objective of this research was to develop a forward artificial neural network to predict 10 year production profile, and to develop an inverse artificial neural network to estimate significant reservoir characterizations, including fracture spacing, porosity and permeability, matrix porosity and permeability, and reservoir pay zone thickness via the commercial software (CMG) and the MATLAB Tool Box. Homogeneous and isotropic double porosity gas reservoirs with dual-lateral wells are the primary research targets.

A total of two thousand dual-porosity reservoir characterization data sets were randomly generated by MATLAB and then fed into CMG software to produce a corresponding 10-year field production profiles. Afterward, 1965 reasonable data sets were selected to build and train the artificial neural networks. For the forward artificial neural network, reservoir characteristics were considered as input while field production rates were considered as output. For the inverse artificial neural network, the aforementioned desired reservoir characteristics mentioned above were selected as network output and the rest of the reservoir characterizations, combined with 3-year production profile, were considered as network input. Last but not the least, two concise graphic user interfaces were established for the user to utilize the two artificial neural network models more conveniently.

The generated artificial neural networks are considered as a reliable model to predict the rate transient data by forward part and to predict matrix porosity, matrix permeability, reservoir thickness, fracture porosity, fracture permeability, and fracture spacing by inverse part for dual-porosity single phase gas reservoirs with dual lateral wells. The analyzed results showed that the forward neural network has approximately 99% probability to generate the rate transient data with less than 1% error. For inverse neural network, nearly 85% probability to predict fracture permeability within 5% error, approximately 91% chance to predict fracture porosity within 5% error, almost 88% probability to predict formation thickness within 5% error, about 68% probability to predict matrix permeability within 5% error, approximately 67% probability to predict matrix porosity within 5% error, and around 43% probability to predict fracture spacing within 10% error. Although the error of fracture spacing is relatively high, the entire quality of the inverse network is considered as reliable. Combined with graphic user interfaces, it is believed that the model developed in the research project was a straightforward, inexpensive, and credible tool for analyzing the rate transient data and for predicting reservoir characterizations for double porosity reservoirs with dual-lateral wells.

This tool can be further enhanced to eliminate the pre-defined design constraints:

1. The produced hydrocarbon can be extended to multi-phase production
2. The fracture characterizations can be extended to anisotropic
3. Skin factor can be added to the analysis
4. The two lateral wells can be placed at the same depth
5. Gravitational effects can be considered
6. The surface drilling location can be randomly placed in the reservoir
7. Phase angle for lateral well pairs can be added for analyzing

References

Alajmi, M. (2007). *The Development of An Artificial Neural Network as A Pressure Transient Analysis Tool for Applications in Double-porosity Reservoirs.* Master's thesis, The Pennsylvania State University.

Bansal, Y. (2009). *Conducting In-situ Combustion Tube Experiments Using Artificial Neural Networks.* Master's thesis. The Pennsylvania State University.

Demuth, H., & Beale (1992). *Neural Network Toolbox for Use with Matlab.* Version 4. Mathworks, Natic, Mass.

Demuth, H., Beale, M., & Hagan, M. (2009). *Neural Network Toolbox™ User's Guide*. The MathWorks, Inc.

Enab, K. (2012). *Artificial Neural Network Based Design for Dual Lateral Well Applications.* University Park, PA. The Pennsylvania State University.

Fractured Reservoirs. (2012). Retrieved from http://www.ireservoir.com/downloads/4_Fractured_Reservoirs_2012Sept.pdf

Kistak, N. (2013). *Development of An Artificial Neural Network for Dual Lateral Horizontal Wells in Gas Reservoirs.* University Park, PA. The Pennsylvania State University.

Kulga, I. (2010). *Development of An Artificial Neural Network for Hydraulically Fractured Horizontal Wells in Tight Gas Sands.* University Park, PA. The Pennsylvania State University.

Lodish, H., Berk, A., & Zipursky, S. L., et al. Molecular Cell Biology. 4th edition. New York: W. H. Freeman; 2000. Section 21.1, *Overview of Neuron Structure and Function.* Available from: http://www.ncbi.nlm.nih.gov/books/NBK21535/

Meiller, M. F. (1993). A Scaled Conjugate Gradient Algorithm for Fast Supervised Learning. *Neural Networks*, 525-533.

Mohaghegh, S. (2000). Applications of Virtual Intelligence to Petroleum Engineering. *Journal of Petroleum Technology 52*: pp. 6473.

Nelson, R. (2001). *Geologic Analysis of Naturally Fractured Reservoirs* (2nd ed.). Houston: Gulf Publishing Company.

Ozkan, E., (1998). Transient Pressure Behavior of Dual Lateral Wells. *SPE*, pp. 181-190.

Streltsova, T. (1988). *Well testing in heterogeneous formations.* New York: Wiley.

Swan Energy Explains Modern Horizontal Drilling Techniques. (2012 March 12). Retrieved from https://swanenergyinc.wordpress.com/tag/horizontal-drilling-techniques/

Vacic, V. (2005). *Summary of The Training Functions in Matlab's NN Toolbox.* University of California. Retrieved from http://alumni.cs.ucr.edu/~vladimir/cs171/nn_summary.pdf

Warren, J. E., & Root, P. J. (1963). The Behavior of Naturally Fractured Reservoirs. *SPEJ* , pp. 245-255.

Appendix A: The Protocol for the ANN Development

A.1 Random Data Generator

```
clear,clc
n=2000; % Number of Cases for CMG Models
A_acre=unifrnd (100,640,[1,n]); % A=Reservoir Area in acre
A_feet=A_acre*43560; % convert to feet^2
R_length=sqrt(A_feet);% Reservoir Length in ft
n_grid=ceil(R_length./120); % grid number with width of 120 ft
D=unifrnd(1000,10000,[1,n]);% D=Reservoir Depth in ft
h=unifrnd(100,400,[1,n]);% Reservoir Thickness in ft
L_h=h./5;% Layer thickness
phi_m=unifrnd(0.08,0.4,[1,n]);% Matrix Porosity
phi_f=unifrnd(0.001,0.03,[1,n]);% Fracture Porosity
k_m=unifrnd(0.001,0.05,[1,n]);% Matrix Permeability
k_f=unifrnd(0.1,1,[1,n]);% Fracture Permeability
fs=unifrnd(1,200,[1,n]);% Fracture Spacing
Cm=unifrnd(0.00000006,0.000003,[1,n]);% Matrix Compressibility
Cf=unifrnd(0.0000006,0.001,[1,n]);% Fracture Compressibility
Theta=unifrnd(0,360,[1,n]);% phase angle
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%need to fix
T=60+D./100;% Reservoir Temperature
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
P=unifrnd(1000,8000,[1,n]);% Reservoir Pressure
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%need to fix
BHP=unifrnd(0,0.5,[1,n]);% BHP=Bottomhole Pressure (psi)
BHP=BHP.*P+14.7;
krg=unifrnd(0.6,0.9,[1,n]);% krg=Maximum Relative Gas Permeability
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% lateral well length
L_L1=unifrnd(500,2000,[1,n]);
L_L2=unifrnd(500,2000,[1,n]);
% lateral well Depth
D_L1=unifrnd(50,350,[1,n]);
D_L2=unifrnd(50,350,[1,n]);

%% Input Check for Well Length (Well Length<0.5Edge of Reservoir)
for i=1:n
    if L_L1(1,i)>0.5*R_length(1,i)
       L_L1(1,i)=unifrnd(500,0.5*R_length(1,i));
    end
    n_L1(1,i)=ceil(L_L1(1,i)/120); %number of blocks for well
    if L_L2(1,i)>0.5*R_length(i)
       L_L2(1,i)=unifrnd(500,0.5*R_length(1,i));
    end
    n_L2(1,i)=ceil(L_L2(1,i)/120); %number of blocks for well
end



%%% Input Check for Well Depth (Well Depth<Thickness of Reservoir)
n_D_L1=ones(1,n);
```

```
n_D_L2=ones(1,n);
for i=1:n
    if D_L1(1,i)>h(1,i)
        D_L1(1,i)=unifrnd(50,h(1,i));
    end

    if D_L2(1,i)>h(1,i)
        D_L2(1,i)=unifrnd(50,h(1,i));
    end
end
n_D_L1=ceil(D_L1./L_h); %%% Determine the layer of wells
n_D_L2=ceil(D_L2./L_h);%%% Determine the layer of wells

%%% put wells in different layers
for i=1:n
    while n_D_L1(1,i)==n_D_L2(1,i)
        D_L1(1,i)=unifrnd(50,350);
        D_L2(1,i)=unifrnd(50,350);
        if D_L1(1,i)>h(1,i)
            D_L1(1,i)=unifrnd(50,h(1,i));
        end
        if D_L2(1,i)>h(1,i)
            D_L2(1,i)=unifrnd(50,h(1,i));
        end
        n_D_L1(1,i)=ceil(D_L1(1,i)/L_h(1,i));
        n_D_L2(1,i)=ceil(D_L2(1,i)/L_h(1,i));
    end
end

% determination of the number of grid blocks for each well lateral
for i=1:n
    Block_n1(1,i)=ceil(L_L1(1,i)/120);
    Block_n2(1,i)=ceil(L_L2(1,i)/120);
end
% determination of center block for vertical well
for i=1:n
    wellblock(1,i)=ceil(n_grid(1,i)/2);
end
%Generate a table
Table=zeros(n,6);
Table(:,1)=D_L1;%column 1 is depth of first lateral well
Table(:,2)=D_L2;%column 2 is depth of second lateral well
Table(:,3)=h;%column 3 is depth of Reservoir
Table(:,4)=L_L1;%column 4 is length of first lateral well
Table(:,5)=L_L2;%column 5 is length of second lateral well
Table(:,6)=R_length;%column 6 is length of Reservoir
%%% Data Compilation
input=[A_feet;n_grid;D;h;L_h;phi_m;phi_f;k_m;k_f;fs;Cm;Cf;Theta;T;P;BHP;krg;L
_L1;L_L2;D_L1;D_L2;wellblock;n_L1;n_L2;n_D_L1;n_D_L2];
input=transpose(input);
%%% Exporting Files
save input.txt input -ASCII
dlmwrite('input.xls', input, 'delimiter', '\t', 'precision', 3)
```

A.2 Simulations Generator

```
%% This file generates input files for CMG (Created by Yogesh Bansal, 2009)
clc
clear
tic
%% Inputs for this code
load input.txt
%% Printing CMG files
forCMG= ('CMGBATCH.bat');
fidbat=fopen(forCMG,'wt');
for i=1:length(input(:,1));
    numb= num2str(i);
    temp=['data' numb '.dat'];
    fid=fopen(temp,'wt');
    fprintf(fid,'\n **Parameters for this run:');
    fprintf(fid,'\n ** 1)A_feet = %g ft',input(i,1));
    fprintf(fid,'\n ** 2)n_grid = %g',input(i,2));
    fprintf(fid,'\n ** 3)D = %g ft',input(i,3));
    fprintf(fid,'\n ** 4)h = %g ft',input(i,4));
    fprintf(fid,'\n ** 5)L_h = %g ft',input(i,5));
    fprintf(fid,'\n ** 6)phi_m = %g',input(i,6));
    fprintf(fid,'\n ** 7)phi_f = %g',input(i,7));
    fprintf(fid,'\n ** 8)k_m = %g mD',input(i,8));
    fprintf(fid,'\n ** 9)k_f = %g mD',input(i,9));
    fprintf(fid,'\n ** 10)fs = %g ft',input(i,10));
    fprintf(fid,'\n ** 11)Cm = %g 1/psi',input(i,11));
    fprintf(fid,'\n ** 12)Cf = %g 1/psi',input(i,12));
    fprintf(fid,'\n ** 13)Theta = %g degrees',input(i,13));
    fprintf(fid,'\n ** 14)T = %g F',input(i,14));
    fprintf(fid,'\n ** 15)P = %g psi',input(i,15));
    fprintf(fid,'\n ** 16)BHP = %g psi',input(i,16));
    fprintf(fid,'\n ** 17)krg = %g',input(i,17));
    %well parameters
    fprintf(fid,'\n ** 18)L_L1 = %g ft',input(i,18));
    fprintf(fid,'\n ** 19)L_L2 = %g ft',input(i,19));
    fprintf(fid,'\n ** 20)D_L1 = %g ft',input(i,20));
    fprintf(fid,'\n ** 21)D_L2 = %g ft',input(i,21));
    fprintf(fid,'\n ** 22)wellblock = %g',input(i,22));
    fprintf(fid,'\n ** 23)n_L1 = %g',input(i,23));
    fprintf(fid,'\n ** 24)n_L2 = %g',input(i,24));
    fprintf(fid,'\n ** 25)n_D_L1 = %g',input(i,25));
    fprintf(fid,'\n ** 26)n_D_L2 = %g',input(i,26));


    fprintf(fid,'\n **');
    fprintf(fid,'\n **');
    fprintf(fid,'\n RESULTS SIMULATOR IMEX 201211');
    fprintf(fid,'\n');
    fprintf(fid,'\n INUNIT FIELD');
    fprintf(fid,'\n WSRF WELL 1');
    fprintf(fid,'\n WSRF GRID TIME');
    fprintf(fid,'\n WSRF SECTOR TIME');
    fprintf(fid,'\n OUTSRF WELL LAYER NONE');
```

```
    fprintf(fid,'\n OUTSRF RES ALL');
    fprintf(fid,'\n OUTSRF GRID SO SG SW PRES OILPOT BPP SSPRES WINFLUX ');
    fprintf(fid,'\n WPRN GRID 0');
    fprintf(fid,'\n OUTPRN GRID NONE');
    fprintf(fid,'\n OUTPRN RES NONE');
    fprintf(fid,'\n **$ Distance units: ft');
    fprintf(fid,'\n RESULTS XOFFSET 0.0000');
    fprintf(fid,'\n RESULTS YOFFSET 0.0000');
    fprintf(fid,'\n RESULTS ROTATION 0.0000 **$ (DEGREES)');
    fprintf(fid,'\n RESULTS AXES-DIRECTIONS 1.0 -1.0 1.0');
    fprintf(fid,'\n
**$ *************************************************************');
    fprintf(fid,'\n **$ Definition of fundamental cartesian grid');
    fprintf(fid,'\n
**$ *************************************************************');
    fprintf(fid,'\n GRID VARI %d %d 5',input(i,2),input(i,2));
    fprintf(fid,'\n KDIR DOWN');
    fprintf(fid,'\n DI IVAR');
    fprintf(fid,'\n %d*120',input(i,2));
    fprintf(fid,'\n DJ JVAR');
    fprintf(fid,'\n %d*120',input(i,2));
    fprintf(fid,'\n DK ALL');
    fprintf(fid,'\n %d*%d',input(i,2)*input(i,2)*5,input(i,4));
    fprintf(fid,'\n DTOP');
    fprintf(fid,'\n %d*0',input(i,2)*input(i,2));
    fprintf(fid,'\n DUALPOR');
    fprintf(fid,'\n SHAPE WR');
    fprintf(fid,'\n **$ 0 = null block, 1 = active block');
    fprintf(fid,'\n NULL *MATRIX CON          1');
    fprintf(fid,'\n **$ 0 = null block, 1 = active block');
    fprintf(fid,'\n NULL *FRACTURE CON        1');
    fprintf(fid,'\n POR *MATRIX CON        %d',input(i,6));
    fprintf(fid,'\n POR *FRACTURE CON      %d',input(i,7));
    fprintf(fid,'\n PERMI *MATRIX CON      %d',input(i,8));
    fprintf(fid,'\n PERMI *FRACTURE CON    %d',input(i,9));
    fprintf(fid,'\n PERMJ *MATRIX CON      %d',input(i,8));
    fprintf(fid,'\n PERMJ *FRACTURE CON    %d',input(i,9));
    fprintf(fid,'\n PERMK *MATRIX CON      %d',input(i,8));
    fprintf(fid,'\n PERMK *FRACTURE CON    %d',input(i,9));
    fprintf(fid,'\n DJFRAC CON       %d',input(i,10));
    fprintf(fid,'\n DIFRAC CON       %d',input(i,10));
    fprintf(fid,'\n DKFRAC CON       %d',input(i,10));
    fprintf(fid,'\n **$ 0 = pinched block, 1 = active block');
    fprintf(fid,'\n PINCHOUTARRAY CON         1');
    fprintf(fid,'\n CPOR FRACTURE %d',input(i,12));
    fprintf(fid,'\n CPOR MATRIX %d',input(i,11));
    fprintf(fid,'\n MODEL GASWATER');
    fprintf(fid,'\n TRES %d',input(i,14));
    fprintf(fid,'\n PVTG EG 1');
    fprintf(fid,'\n **$             p        Eg        visg');
    fprintf(fid,'\n        14.696   5.14709  0.0120493');
    fprintf(fid,'\n        347.05   125.961  0.0123771');
    fprintf(fid,'\n       679.403   255.201  0.0128714');
    fprintf(fid,'\n       1344.11   535.365  0.0142606');
    fprintf(fid,'\n       1676.46   681.686  0.0151437');
    fprintf(fid,'\n       2008.82   827.424   0.016133');
    fprintf(fid,'\n       2341.17    968.74  0.0172027');
```

```
    fprintf(fid,'\n       2673.52   1102.58  0.0183231');
    fprintf(fid,'\n       3005.88   1227.09  0.0194662');
    fprintf(fid,'\n       3338.23   1341.56  0.0206092');
    fprintf(fid,'\n       3670.59   1446.05  0.0217354');
    fprintf(fid,'\n       4002.94   1541.15  0.0228338');
    fprintf(fid,'\n       4335.29   1627.64  0.0238974');
    fprintf(fid,'\n       4667.65   1706.42  0.0249228');
    fprintf(fid,'\n          5000   1778.33  0.0259088');
    fprintf(fid,'\n GRAVITY GAS 0.554');
    fprintf(fid,'\n REFPW 14.696');
    fprintf(fid,'\n DENSITY WATER 62.3362');
    fprintf(fid,'\n BWI 1.00803');
    fprintf(fid,'\n CW 3.17118e-006');
    fprintf(fid,'\n VWI 0.69308');
    fprintf(fid,'\n CVW 0');
    fprintf(fid,'\n ROCKFLUID');
    fprintf(fid,'\n RPT');
    fprintf(fid,'\n **$         Sw          krw');
    fprintf(fid,'\n SWT                        ');
    fprintf(fid,'\n               0            0');
    fprintf(fid,'\n          0.0625  0.000390625');
    fprintf(fid,'\n           0.125    0.0015625');
    fprintf(fid,'\n          0.1875   0.00351563');
    fprintf(fid,'\n            0.25      0.00625');
    fprintf(fid,'\n          0.3125   0.00976563');
    fprintf(fid,'\n           0.375    0.0140625');
    fprintf(fid,'\n          0.4375    0.0191406');
    fprintf(fid,'\n             0.5        0.025');
    fprintf(fid,'\n          0.5625    0.0316406');
    fprintf(fid,'\n           0.625    0.0390625');
    fprintf(fid,'\n          0.6875    0.0472656');
    fprintf(fid,'\n            0.75      0.05625');
    fprintf(fid,'\n          0.8125    0.0660156');
    fprintf(fid,'\n           0.875    0.0765625');
    fprintf(fid,'\n          0.9375    0.0878906');
    fprintf(fid,'\n               1          0.1');
    fprintf(fid,'\n **$         Sl          krg');
    fprintf(fid,'\n SLT                        ');
    fprintf(fid,'\n               0          0.9');
    fprintf(fid,'\n         0.01875     0.791016');
    fprintf(fid,'\n          0.0375     0.689063');
    fprintf(fid,'\n         0.05625     0.594141');
    fprintf(fid,'\n           0.075      0.50625');
    fprintf(fid,'\n         0.09375     0.425391');
    fprintf(fid,'\n          0.1125     0.351562');
    fprintf(fid,'\n         0.13125     0.284766');
    fprintf(fid,'\n            0.15        0.225');
    fprintf(fid,'\n         0.16875     0.172266');
    fprintf(fid,'\n          0.1875     0.126563');
    fprintf(fid,'\n         0.20625    0.0878906');
    fprintf(fid,'\n           0.225      0.05625');
    fprintf(fid,'\n         0.24375    0.0316406');
    fprintf(fid,'\n          0.2625    0.0140625');
    fprintf(fid,'\n         0.28125   0.00351563');
    fprintf(fid,'\n             0.3            0');
    fprintf(fid,'\n            0.55            0');
    fprintf(fid,'\n             0.8            0');
```

```
    fprintf(fid,'\n INITIAL');
    fprintf(fid,'\n USER_INPUT');
    fprintf(fid,'\n');
    fprintf(fid,'\n GOC_PC 0');
    fprintf(fid,'\n WOC_PC 0');
    fprintf(fid,'\n PRES *MATRIX CON        %d',input(i,15));
    fprintf(fid,'\n PRES *FRACTURE CON        %d',input(i,15));
    fprintf(fid,'\n SW *MATRIX CON          0');
    fprintf(fid,'\n SW *FRACTURE CON          0');
    fprintf(fid,'\n NUMERICAL');
    fprintf(fid,'\n RUN');
    fprintf(fid,'\n DATE 2015 1 1');
    fprintf(fid,'\n');
    fprintf(fid,'\n **$');
    fprintf(fid,'\n WELL ''Well-1''');
    fprintf(fid,'\n PRODUCER ''Well-1''');
    fprintf(fid,'\n OPERATE MIN BHP %d CONT',input(i,16));
    fprintf(fid,'\n **$        rad geofac wfrac skin');
    fprintf(fid,'\n GEOMETRY K 0.3 0.37 1. 0.');
    fprintf(fid,'\n PERF GEOA ''Well-1''');
    fprintf(fid,'\n **$ UBA ff Status Connection ');
    % well 1
    fprintf(fid,'\n     %d    %d   %d   1.   OPEN    FLOW-TO   ''SURFACE''
REFLAYER',input(i,22),input(i,22),input(i,25));
    for L1=1:input(i,23)
        fprintf(fid,'\n     %d    %d   %d   1.   OPEN    FLOW-
TO  %d',input(i,22)+L1,input(i,22),input(i,25),L1);
    end

    fprintf(fid,'\n **$');
    fprintf(fid,'\n WELL ''Well-2''');
    fprintf(fid,'\n PRODUCER ''Well-2''');
    fprintf(fid,'\n OPERATE MIN BHP %d CONT',input(i,16));
    fprintf(fid,'\n **$        rad geofac wfrac skin');
    fprintf(fid,'\n GEOMETRY K 0.3 0.37 1. 0.');
    fprintf(fid,'\n PERF GEOA ''Well-2''');
    fprintf(fid,'\n **$ UBA ff Status Connection ');
    % well 2
    fprintf(fid,'\n     %d    %d   %d   1.   OPEN    FLOW-TO   ''SURFACE''
REFLAYER',input(i,22),input(i,22),input(i,26));
    for L2=1:input(i,24)
        fprintf(fid,'\n     %d    %d   %d   1.   OPEN    FLOW-
TO  %d',input(i,22)+L2,input(i,22),input(i,26),L2);
    end
    fprintf(fid,'\n DATE 2015 2 1.00000');
    fprintf(fid,'\n DATE 2015 3 1.00000');
    fprintf(fid,'\n DATE 2015 4 1.00000');
    fprintf(fid,'\n DATE 2015 5 1.00000');
    fprintf(fid,'\n DATE 2015 6 1.00000');
    fprintf(fid,'\n DATE 2015 7 1.00000');
    fprintf(fid,'\n DATE 2015 8 1.00000');
    fprintf(fid,'\n DATE 2015 9 1.00000');
    fprintf(fid,'\n DATE 2015 10 1.00000');
    fprintf(fid,'\n DATE 2015 11 1.00000');
    fprintf(fid,'\n DATE 2015 12 1.00000');
    fprintf(fid,'\n DATE 2016 1 1.00000');
```

```
    fprintf(fid,'\n DATE 2016 2 1.00000');
    fprintf(fid,'\n DATE 2016 3 1.00000');
    fprintf(fid,'\n DATE 2016 4 1.00000');
    fprintf(fid,'\n DATE 2016 5 1.00000');
    fprintf(fid,'\n DATE 2016 6 1.00000');
    fprintf(fid,'\n DATE 2016 7 1.00000');
    fprintf(fid,'\n DATE 2016 8 1.00000');
    fprintf(fid,'\n DATE 2016 9 1.00000');
    fprintf(fid,'\n DATE 2016 10 1.00000');
    fprintf(fid,'\n DATE 2016 11 1.00000');
    fprintf(fid,'\n DATE 2016 12 1.00000');
    fprintf(fid,'\n DATE 2017 1 1.00000');
    fprintf(fid,'\n DATE 2017 2 1.00000');
    fprintf(fid,'\n DATE 2017 3 1.00000');
    fprintf(fid,'\n DATE 2017 4 1.00000');
    fprintf(fid,'\n DATE 2017 5 1.00000');
    fprintf(fid,'\n DATE 2017 6 1.00000');
    fprintf(fid,'\n DATE 2017 7 1.00000');
    fprintf(fid,'\n DATE 2017 8 1.00000');
    fprintf(fid,'\n DATE 2017 9 1.00000');
    fprintf(fid,'\n DATE 2017 10 1.00000');
    fprintf(fid,'\n DATE 2017 11 1.00000');
    fprintf(fid,'\n DATE 2017 12 1.00000');
    fprintf(fid,'\n DATE 2018 1 1.00000');
    fprintf(fid,'\n DATE 2018 2 1.00000');
    fprintf(fid,'\n DATE 2018 3 1.00000');
    fprintf(fid,'\n DATE 2018 4 1.00000');
    fprintf(fid,'\n DATE 2018 5 1.00000');
    fprintf(fid,'\n DATE 2018 6 1.00000');
    fprintf(fid,'\n DATE 2018 7 1.00000');
    fprintf(fid,'\n DATE 2018 8 1.00000');
    fprintf(fid,'\n DATE 2018 9 1.00000');
    fprintf(fid,'\n DATE 2018 10 1.00000');
    fprintf(fid,'\n DATE 2018 11 1.00000');
    fprintf(fid,'\n DATE 2018 12 1.00000');
    fprintf(fid,'\n DATE 2019 1 1.00000');
    fprintf(fid,'\n DATE 2019 2 1.00000');
    fprintf(fid,'\n DATE 2019 3 1.00000');
    fprintf(fid,'\n DATE 2019 4 1.00000');
    fprintf(fid,'\n DATE 2019 5 1.00000');
    fprintf(fid,'\n DATE 2019 6 1.00000');
    fprintf(fid,'\n DATE 2019 7 1.00000');
    fprintf(fid,'\n DATE 2019 8 1.00000');
    fprintf(fid,'\n DATE 2019 9 1.00000');
    fprintf(fid,'\n DATE 2019 10 1.00000');
    fprintf(fid,'\n DATE 2019 11 1.00000');
    fprintf(fid,'\n DATE 2019 12 1.00000');
    fprintf(fid,'\n DATE 2020 1 1.00000');
    fprintf(fid,'\n DATE 2020 2 1.00000');
    fprintf(fid,'\n DATE 2020 3 1.00000');
    fprintf(fid,'\n DATE 2020 4 1.00000');
    fprintf(fid,'\n DATE 2020 5 1.00000');
    fprintf(fid,'\n DATE 2020 6 1.00000');
    fprintf(fid,'\n DATE 2020 7 1.00000');
    fprintf(fid,'\n DATE 2020 8 1.00000');
    fprintf(fid,'\n DATE 2020 9 1.00000');
    fprintf(fid,'\n DATE 2020 10 1.00000');
```

```
    fprintf(fid,'\n DATE 2020 11 1.00000');
    fprintf(fid,'\n DATE 2020 12 1.00000');
    fprintf(fid,'\n DATE 2021 1 1.00000');
    fprintf(fid,'\n DATE 2021 2 1.00000');
    fprintf(fid,'\n DATE 2021 3 1.00000');
    fprintf(fid,'\n DATE 2021 4 1.00000');
    fprintf(fid,'\n DATE 2021 5 1.00000');
    fprintf(fid,'\n DATE 2021 6 1.00000');
    fprintf(fid,'\n DATE 2021 7 1.00000');
    fprintf(fid,'\n DATE 2021 8 1.00000');
    fprintf(fid,'\n DATE 2021 9 1.00000');
    fprintf(fid,'\n DATE 2021 10 1.00000');
    fprintf(fid,'\n DATE 2021 11 1.00000');
    fprintf(fid,'\n DATE 2021 12 1.00000');
    fprintf(fid,'\n DATE 2022 1 1.00000');
    fprintf(fid,'\n DATE 2022 2 1.00000');
    fprintf(fid,'\n DATE 2022 3 1.00000');
    fprintf(fid,'\n DATE 2022 4 1.00000');
    fprintf(fid,'\n DATE 2022 5 1.00000');
    fprintf(fid,'\n DATE 2022 6 1.00000');
    fprintf(fid,'\n DATE 2022 7 1.00000');
    fprintf(fid,'\n DATE 2022 8 1.00000');
    fprintf(fid,'\n DATE 2022 9 1.00000');
    fprintf(fid,'\n DATE 2022 10 1.00000');
    fprintf(fid,'\n DATE 2022 11 1.00000');
    fprintf(fid,'\n DATE 2022 12 1.00000');
    fprintf(fid,'\n DATE 2023 1 1.00000');
    fprintf(fid,'\n DATE 2023 2 1.00000');
    fprintf(fid,'\n DATE 2023 3 1.00000');
    fprintf(fid,'\n DATE 2023 4 1.00000');
    fprintf(fid,'\n DATE 2023 5 1.00000');
    fprintf(fid,'\n DATE 2023 6 1.00000');
    fprintf(fid,'\n DATE 2023 7 1.00000');
    fprintf(fid,'\n DATE 2023 8 1.00000');
    fprintf(fid,'\n DATE 2023 9 1.00000');
    fprintf(fid,'\n DATE 2023 10 1.00000');
    fprintf(fid,'\n DATE 2023 11 1.00000');
    fprintf(fid,'\n DATE 2023 12 1.00000');
    fprintf(fid,'\n DATE 2024 1 1.00000');
    fprintf(fid,'\n DATE 2024 2 1.00000');
    fprintf(fid,'\n DATE 2024 3 1.00000');
    fprintf(fid,'\n DATE 2024 4 1.00000');
    fprintf(fid,'\n DATE 2024 5 1.00000');
    fprintf(fid,'\n DATE 2024 6 1.00000');
    fprintf(fid,'\n DATE 2024 7 1.00000');
    fprintf(fid,'\n DATE 2024 8 1.00000');
    fprintf(fid,'\n DATE 2024 9 1.00000');
    fprintf(fid,'\n DATE 2024 10 1.00000');
    fprintf(fid,'\n DATE 2024 11 1.00000');
    fprintf(fid,'\n DATE 2024 12 1.00000');
    fprintf(fid,'\n DATE 2025 1 1.00000');
    fprintf(fid,'\n *stop');
    fprintf(fid,'\n **');
    fprintf(fid,'\n **');
    fclose(fid);
    fprintf(fidbat,'%s','call "C:\\Program Files
(x86)\\CMG\\IMEX\\2012.11\\Win_x64\\EXE\\MX201211.exe" -f data');
```

```
    fprintf(fidbat,num2str(i));
    fprintf(fidbat,'.dat\n');
end
fclose(fidbat);
toc
```

A.3 Results Extractor

```
%% This file extracts monthly production data from CMG (Created by Yogesh
Bansal, 2009)
clc
clear
tic
%% Inputs for this code
load input.txt
%% Extracts Data from CMG Results
fordata=('CMGRWD.bat');
fidrwd=fopen(fordata,'wt');
for i=1:length(input(:,1));
    numb = num2str(i);
    dataext = ['data' numb '.rwd' ];
    fid = fopen(dataext,'wt+');
    numb = num2str(i);
    fprintf(fid,'%s','FILE ''data',numb);
    fprintf(fid,'%s','.irf''');
    fprintf(fid,'SPREADSHEET\n');
    fprintf(fid,'TABLE-FOR\n');
    %fprintf(fid,'%s\n','COLUMN-FOR *PARAMETERS ''Cumulative Gas SC'' *WELLS
''Well-1''');
    fprintf(fid,'%s\n','COLUMN-FOR *PARAMETERS ''Gas Rate SC - Daily'' *WELLS
''Well-1''');
    %fprintf(fid,'%s\n','COLUMN-FOR *PARAMETERS ''Cumulative Gas SC''  *WELLS
''Well-2''');
    fprintf(fid,'%s\n','COLUMN-FOR *PARAMETERS ''Gas Rate SC - Daily'' *WELLS
''Well-2''');
    %fprintf(fid,'%s\n','COLUMN-FOR *PARAMETERS ''Cumulative Gas SC'' *GROUPS
''Default-Field-PRO''');
    fprintf(fid,'%s\n','COLUMN-FOR *PARAMETERS ''Gas Rate SC - Daily''
*GROUPS ''Default-Field-PRO''');
    fprintf(fid,'%s\n','TABLE-END');
    fclose(fid);
    fprintf(fidrwd,'%s','call "C:\\Program Files
(x86)\\CMG\\BR\\2012.20\\Win_x64\\EXE\\report.exe" -f "data');
    fprintf(fidrwd,num2str(i));
    fprintf(fidrwd,'%s','.rwd"');
    fprintf(fidrwd,'%s',' -o "data',numb,'.txt"');
    fprintf(fidrwd,'\n');
end
fclose(fidrwd);
toc
```

A.4 Data Selection

```
%% Data Selection Monthly Production Data
clear,clc
%% Inputs for this code
load input.txt
%% Collecting data from CMG results
count=0;
for i=1:length(input(:,1));
    run = num2str(i);
    data = ['data' run '.txt' ];
    [time,scfd1,scfd2,scfdfield] =
textread(data,'%f %f %f %f','headerlines',6);
    k=1;
    for j=2:60:length(time)
        q(i,k) = scfdfield(j,1);
        k=k+1;
    end
end
k=1;
for j=2:60:length(time)
    t(1,k)=time(j,1);
    k=k+1;
end
```

A.5 Forward ANN Development

```
%% ANN Training
clc,clear
format long
%% Importing data
input=xlsread('Traininginput_test', 1, 'A1:Q1965');
output=xlsread('Trainingoutput_test', 1, 'A1:AE1965');
%% Training
P=transpose(input);
T=transpose(output);
%Normalizing the data
%Pn stands for normalized input and Tn stands for normalized output
[Pn,ps]=mapminmax(P,-1,1); % gives all values between -1 & 1
[Tn,ts]=mapminmax(T,-1,1); % gives all values between -1 & 1
[mi,ni]=size(Pn);
[mo,no]=size(Tn);
%defining some variables required in the network
N_in=mi; % number of inputs in the network
N_out=mo; % number of outputs in the network
Tot_in =ni; %total no. of simulations
%seperating training, testing & validation data when random
%selection command is available through higher version dividing randomly
[Pn_train,Pn_val,Pn_test,trainInd,valInd,testInd] =
dividerand(Pn,0.7,0.15,0.15);
```

```
[Tn_train,Tn_val,Tn_test] = divideind(Tn,trainInd,valInd,testInd);
val.T=Tn_val;
val.P=Pn_val;
test.T=Tn_test;
test.P=Pn_test;
%Initiating network parameters
NNeu1=25;
NNeu2=25;
NNeu3=20;
%NNeu4=20;
% creating the cascade backpropagation network
net=newcf(Pn,Tn,[NNeu1,NNeu2,NNeu3],{'tansig','tansig','tansig','purelin'},'t
rainscg','learngdm','msereg');
%net=newcf(Pn,Tn,[NNeu1,NNeu2,NNeu3],{'tansig','tansig','tansig'},'trainscg',
'learngdm','msereg');
%net=newcf(Pn,Tn,[NNeu1,NNeu2,NNeu3,NNeu4],{'tansig','tansig','tansig','tansi
g'},'trainscg','learngdm','msereg');
%setting training parameters for the network
net.trainParam.goal=0.0001; %accuracy within this range
net.trainParam.epochs=10000; % number of iteration sets
net.trainParam.show=1;
net.trainParam.max_fail=10000;
net.efficiency.memoryReduction=1; %to reduce memory requirements
net.trainParam.showWindow=true;
%starting training the network
[net,tr] = train(net,Pn_train,Tn_train,[],[],test,val);
%getting data from the trained network
Tn_train_ann=sim(net,Pn_train);
Tn_test_ann=sim(net,Pn_test);
%denormalizing the data sets obtained
%output reversal
T_train=mapminmax('reverse',Tn_train,ts);
T_test=mapminmax('reverse',Tn_test,ts);
T_train_ann=mapminmax('reverse',Tn_train_ann,ts);
T_test_ann=mapminmax('reverse',Tn_test_ann,ts);
%input reversal
Pn_train=mapminmax('reverse',Pn_train,ps);
Pn_val=mapminmax('reverse',Pn_val,ps);
Pn_test=mapminmax('reverse',Pn_test,ps);

error=abs((T_test-T_test_ann)./T_test);
convg=mean(mean(error))
```

Appendix B: Sample Forward ANN Results

	Figure 5.4			Figure 5.5			Figure 5.6		
time	CMG	ANN	Error	CMG	ANN	Error	CMG	ANN	Error
1	2.17E+08	2.03E+08	6.43%	8.74E+08	7.93E+08	9.25%	7.93E+08	9.58E+08	20.86%
121	1.12E+08	1.10E+08	1.78%	3.62E+08	3.22E+08	10.79%	3.81E+08	4.67E+08	22.29%
241	1.03E+08	9.74E+07	5.66%	3.33E+08	3.15E+08	5.34%	3.58E+08	4.15E+08	15.72%
361	9.67E+07	9.20E+07	4.85%	3.1E+08	2.82E+08	8.97%	3.34E+08	3.5E+08	4.90%
481	9.24E+07	9.18E+07	0.59%	2.94E+08	2.74E+08	6.87%	3.12E+08	3.52E+08	12.82%
601	8.91E+07	8.73E+07	2.02%	2.82E+08	2.58E+08	8.66%	2.92E+08	3.32E+08	13.89%
721	8.66E+07	8.40E+07	3.03%	2.73E+08	2.51E+08	8.02%	2.73E+08	2.71E+08	0.60%
841	8.46E+07	8.32E+07	1.74%	2.65E+08	2.42E+08	8.93%	2.55E+08	2.8E+08	9.73%
961	8.29E+07	8.13E+07	1.93%	2.59E+08	2.4E+08	7.21%	2.38E+08	2.52E+08	5.96%
1081	8.14E+07	8.02E+07	1.54%	2.53E+08	2.31E+08	8.52%	2.22E+08	2.24E+08	0.84%
1201	8.01E+07	7.67E+07	4.29%	2.48E+08	2.32E+08	6.23%	2.08E+08	2.09E+08	0.61%
1321	7.89E+07	8.00E+07	1.33%	2.43E+08	2.25E+08	7.51%	1.94E+08	1.93E+08	0.46%
1441	7.78E+07	7.78E+07	0.11%	2.39E+08	2.18E+08	8.43%	1.81E+08	1.77E+08	1.96%
1561	7.69E+07	7.62E+07	0.83%	2.34E+08	2.21E+08	5.79%	1.69E+08	1.57E+08	6.71%
1681	7.59E+07	7.62E+07	0.41%	2.31E+08	2.17E+08	5.74%	1.57E+08	1.51E+08	4.30%
1801	7.51E+07	7.39E+07	1.55%	2.27E+08	2.15E+08	5.37%	1.47E+08	1.33E+08	9.16%
1921	7.42E+07	7.49E+07	0.88%	2.23E+08	2.1E+08	6.02%	1.37E+08	1.31E+08	3.82%
2041	7.34E+07	7.40E+07	0.80%	2.2E+08	2.1E+08	4.70%	1.27E+08	1.19E+08	6.65%
2161	7.28E+07	7.18E+07	1.44%	2.18E+08	2.07E+08	5.10%	1.21E+08	1.1E+08	8.74%
2281	7.21E+07	7.12E+07	1.29%	2.15E+08	1.99E+08	7.44%	1.13E+08	1.08E+08	4.07%
2401	7.14E+07	7.05E+07	1.26%	2.12E+08	2.04E+08	3.79%	1.06E+08	97432574	7.69%
2521	7.07E+07	7.06E+07	0.13%	2.09E+08	1.91E+08	8.42%	98631700	89007704	9.76%
2641	7.00E+07	6.97E+07	0.43%	2.06E+08	1.94E+08	5.91%	92274700	79621885	13.71%
2761	6.94E+07	6.93E+07	0.11%	2.03E+08	1.89E+08	6.67%	86376400	74421489	13.84%
2881	6.87E+07	7.01E+07	1.95%	2E+08	1.88E+08	6.07%	81002500	73524396	9.23%
3001	6.81E+07	6.82E+07	0.17%	1.98E+08	1.86E+08	6.02%	75956200	67987173	10.49%
3121	6.75E+07	6.74E+07	0.10%	1.95E+08	1.81E+08	7.32%	71237600	66709815	6.36%
3241	6.69E+07	6.51E+07	2.68%	1.92E+08	1.82E+08	5.33%	66912300	57772299	13.66%
3361	6.63E+07	6.64E+07	0.11%	1.9E+08	1.75E+08	8.04%	62783500	54701320	12.87%
3481	6.58E+07	6.58E+07	0.02%	1.87E+08	1.7E+08	9.54%	58982400	49307867	16.40%
3601	6.52E+07	6.59E+07	1.07%	1.85E+08	1.77E+08	4.59%	55377900	43418556	21.60%

	Figure 5.7			Figure 5.8		
time	CMG	ANN	Error	CMG	ANN	Error
1	3.41E+09	4.08E+09	19.88%	1.21E+09	1.19E+09	1.13%
121	1.19E+09	1.36E+09	14.38%	4.59E+08	4.73E+08	3.00%
241	1.03E+09	1.2E+09	16.25%	4.17E+08	4.28E+08	2.46%
361	8.83E+08	9.63E+08	9.05%	3.84E+08	3.98E+08	3.76%
481	7.64E+08	8.57E+08	12.12%	3.61E+08	3.69E+08	2.25%
601	6.65E+08	8.08E+08	21.50%	3.42E+08	3.5E+08	2.29%
721	5.84E+08	7.27E+08	24.47%	3.26E+08	3.24E+08	0.80%
841	5.18E+08	6.58E+08	27.14%	3.13E+08	3.2E+08	2.40%
961	4.61E+08	6.38E+08	38.50%	3E+08	3.06E+08	1.91%
1081	4.13E+08	5.83E+08	41.11%	2.89E+08	2.88E+08	0.45%
1201	3.73E+08	4.85E+08	30.05%	2.79E+08	2.83E+08	1.50%
1321	3.38E+08	5.12E+08	51.50%	2.69E+08	2.75E+08	2.30%
1441	3.08E+08	4.15E+08	34.77%	2.6E+08	2.64E+08	1.48%
1561	2.82E+08	3.69E+08	30.83%	2.52E+08	2.5E+08	0.75%
1681	2.59E+08	3.66E+08	41.43%	2.44E+08	2.45E+08	0.52%
1801	2.38E+08	3.15E+08	32.45%	2.36E+08	2.35E+08	0.40%
1921	2.2E+08	2.82E+08	28.25%	2.29E+08	2.34E+08	1.95%
2041	2.04E+08	2.67E+08	31.31%	2.22E+08	2.3E+08	3.38%
2161	1.93E+08	2.54E+08	31.92%	2.18E+08	2.22E+08	1.97%
2281	1.79E+08	2.41E+08	34.63%	2.11E+08	2.13E+08	0.81%
2401	1.67E+08	2.35E+08	40.71%	2.06E+08	2.06E+08	0.00%
2521	1.56E+08	2.02E+08	29.43%	2E+08	2.03E+08	1.75%
2641	1.46E+08	2.23E+08	52.86%	1.95E+08	1.98E+08	1.79%
2761	1.36E+08	2.07E+08	52.04%	1.89E+08	1.9E+08	0.09%
2881	1.28E+08	1.85E+08	44.92%	1.84E+08	1.82E+08	1.53%
3001	1.2E+08	1.72E+08	43.45%	1.8E+08	1.84E+08	2.36%
3121	1.13E+08	1.57E+08	38.85%	1.75E+08	1.82E+08	3.92%
3241	1.06E+08	1.5E+08	41.04%	1.71E+08	1.73E+08	1.52%
3361	1E+08	1.46E+08	46.29%	1.66E+08	1.74E+08	4.34%
3481	94502900	1.47E+08	55.31%	1.62E+08	1.63E+08	0.37%
3601	89129000	1.47E+08	64.99%	1.58E+08	1.64E+08	3.34%

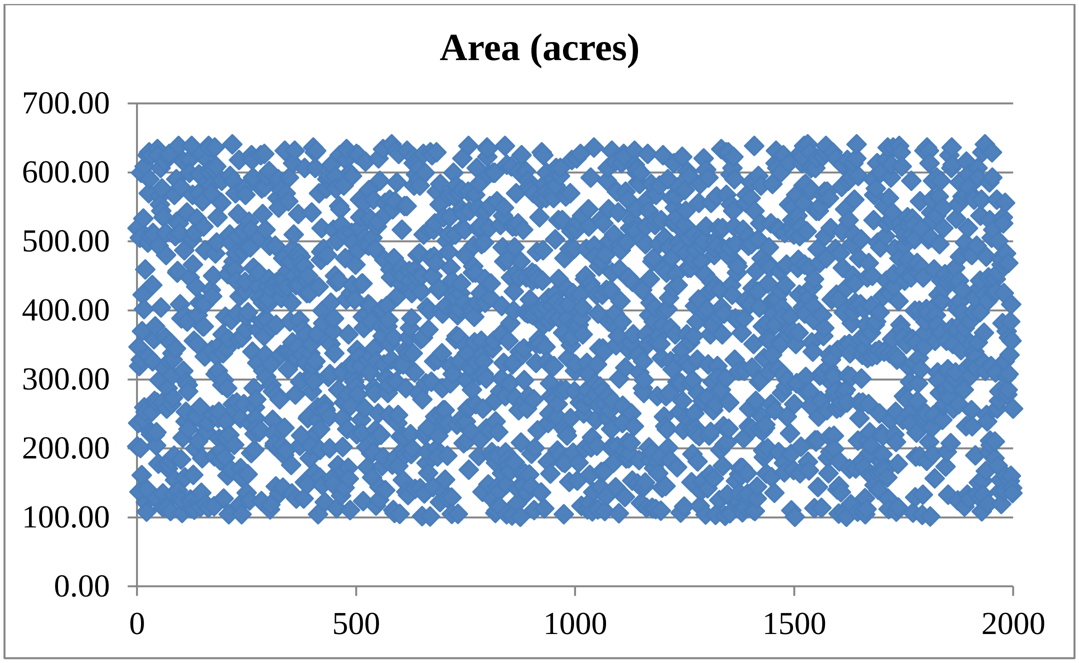
Appendix C: Sample Inverse ANN Test Results

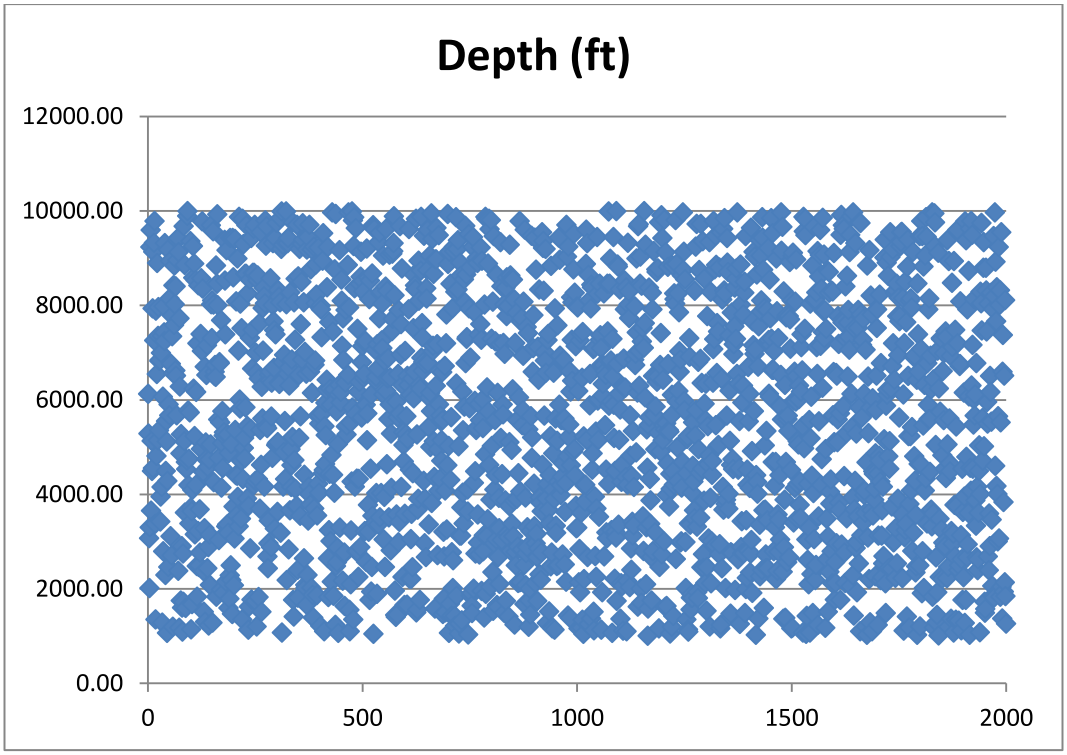
CMG		ANN		Absolute Difference	
Thickness (ft)	Fracture spacing (ft)	Thickness (ft)	Fracture spacing (ft)	Thickness (ft)	Fracture spacing (ft)
322.00	104.00	275.58	67.11	46.42	36.89
243.00	113.00	249.56	112.56	6.56	0.44
115.00	155.00	138.75	170.00	23.75	15.00
192.00	24.20	201.91	52.96	9.91	28.76
114.00	73.20	150.38	141.54	36.38	68.34
344.00	54.10	369.70	75.30	25.70	21.20
307.00	79.40	263.60	114.28	43.40	34.88
392.00	178.00	337.21	154.60	54.79	23.40
366.00	136.00	340.67	134.48	25.33	1.52
182.00	65.10	212.88	59.50	30.88	5.60
277.00	39.10	305.63	92.60	28.63	53.50
398.00	138.00	435.05	66.52	37.05	71.48
151.00	142.00	97.41	78.87	53.59	63.13
357.00	114.00	402.18	112.13	45.18	1.87
392.00	140.00	397.03	57.39	5.03	82.61
173.00	86.20	204.60	113.01	31.60	26.81
224.00	179.00	201.13	160.65	22.87	18.35
211.00	98.50	159.56	87.20	51.44	11.30
318.00	83.80	323.96	83.57	5.96	0.23
113.00	19.00	130.47	89.94	17.47	70.94

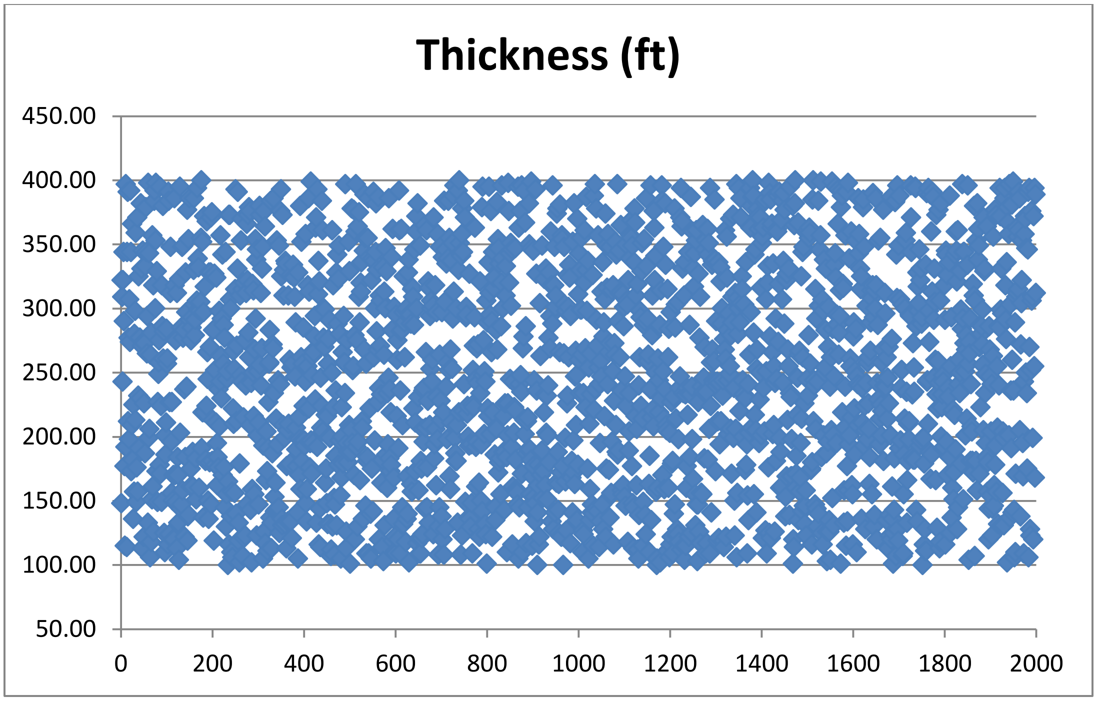
CMG		ANN		Absolute Difference	
Fracture Permeability (mD)	Matrix Porosity	Fracture Permeability (mD)	Matrix Porosity	Fracture Permeability (mD)	Matrix Porosity
0.215	24.70%	0.263	24.58%	0.048	0.12%
0.988	14.80%	0.960	22.62%	0.028	7.82%
0.496	39.90%	0.448	29.37%	0.048	10.53%
0.462	12.90%	0.447	8.34%	0.015	4.56%
0.635	39.10%	0.494	36.39%	0.141	2.71%
0.537	17.00%	0.519	25.17%	0.018	8.17%
0.250	37.60%	0.322	26.95%	0.072	10.65%
0.332	25.70%	0.385	31.28%	0.053	5.58%
0.581	37.60%	0.657	29.66%	0.076	7.94%
0.917	16.50%	0.787	11.68%	0.130	4.82%
0.324	29.40%	0.324	29.13%	0.000	0.27%
0.626	17.70%	0.596	12.98%	0.030	4.72%
0.616	12.90%	0.767	20.01%	0.151	7.11%
0.977	17.10%	0.941	21.38%	0.036	4.28%
0.608	19.50%	0.624	27.89%	0.016	8.39%
0.278	26.20%	0.184	28.53%	0.094	2.33%
0.151	30.90%	0.208	28.42%	0.057	2.48%
0.242	31.90%	0.330	24.97%	0.088	6.93%
0.963	8.21%	0.930	12.73%	0.033	4.52%
0.708	21.50%	0.595	22.71%	0.113	1.21%

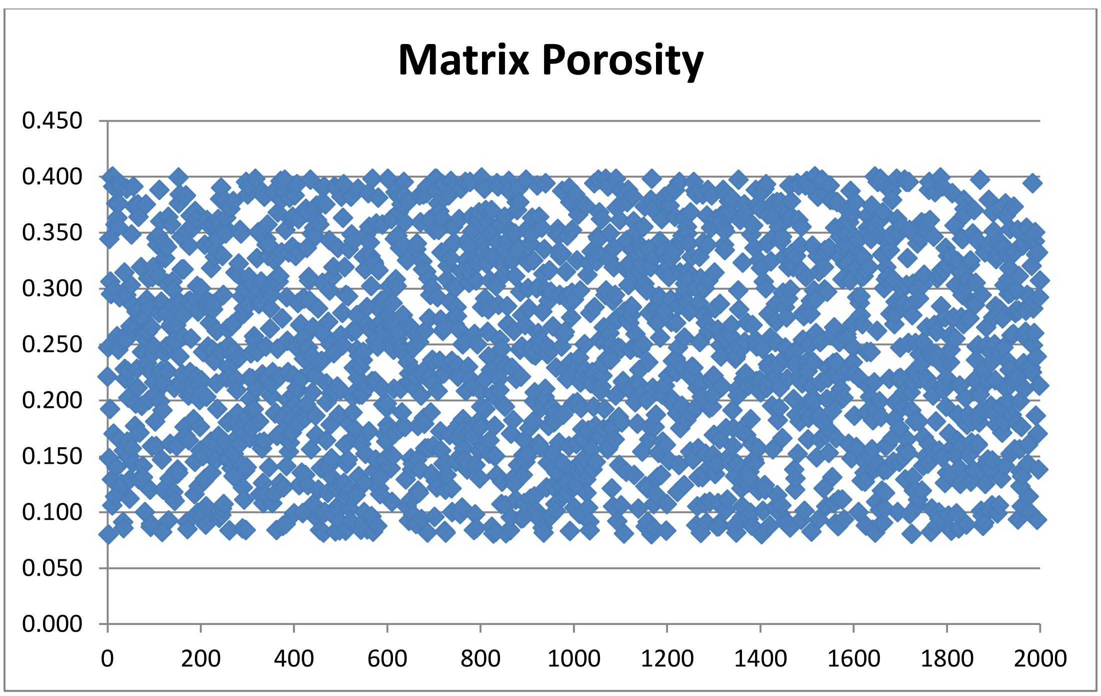
CMG		ANN		Absolute Difference	
Matrix Permeability (mD)	Fracture Porosity	Matrix Permeability (mD)	Fracture Porosity	Matrix Permeability (mD)	Fracture Porosity
0.0315	1.200%	0.02560	1.780%	0.0059	0.58%
0.0486	1.210%	0.03680	0.805%	0.0118	0.41%
0.0450	1.020%	0.01889	1.396%	0.0261	0.38%
0.0035	0.561%	0.02616	1.216%	0.0227	0.65%
0.0047	2.660%	0.01892	1.337%	0.0142	1.32%
0.0023	1.800%	0.02764	1.669%	0.0254	0.13%
0.0310	0.122%	0.02080	1.609%	0.0102	1.49%
0.0181	0.578%	0.02145	1.019%	0.0033	0.44%
0.0360	0.286%	0.01675	1.522%	0.0192	1.24%
0.0492	1.510%	0.02917	1.422%	0.0200	0.09%
0.0214	2.250%	0.03259	1.513%	0.0112	0.74%
0.0426	2.940%	0.02194	1.539%	0.0207	1.40%
0.0103	0.241%	0.01480	1.029%	0.0045	0.79%
0.0222	1.520%	0.03944	0.502%	0.0172	1.02%
0.0206	2.430%	0.02727	2.112%	0.0067	0.32%
0.0087	0.194%	0.02658	1.340%	0.0179	1.15%
0.0115	2.770%	0.01853	1.402%	0.0070	1.37%
0.0479	1.500%	0.02024	1.563%	0.0277	0.06%
0.0122	1.700%	0.02428	1.038%	0.0121	0.66%
0.0376	2.920%	0.02400	1.842%	0.0136	1.08%

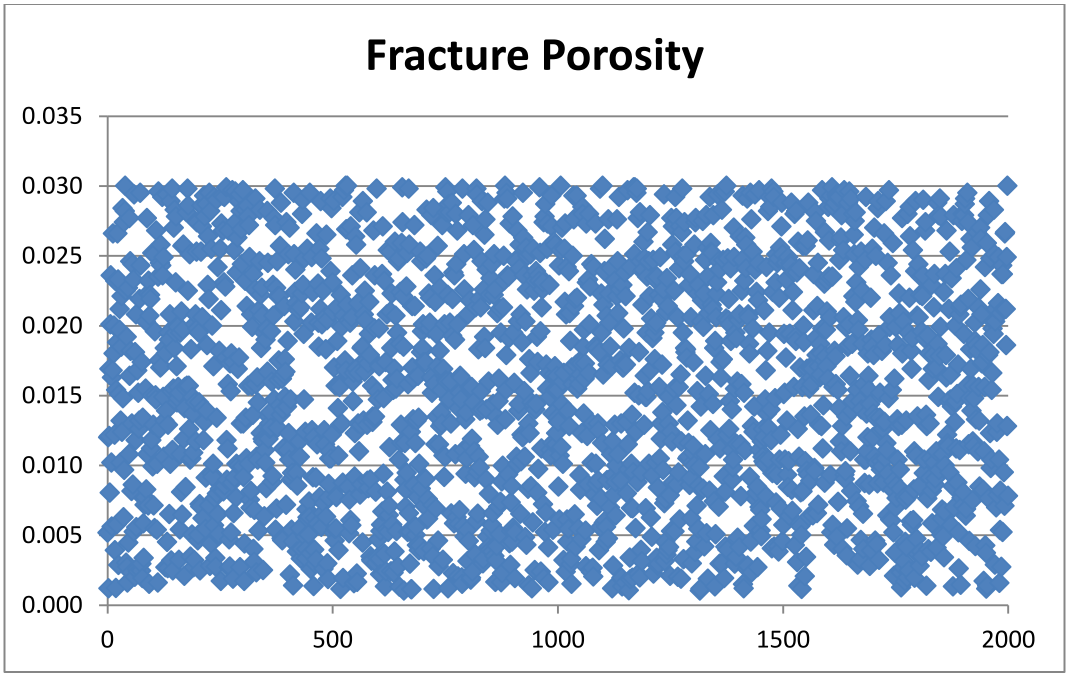
Appendix D: Parameter Distribution Maps

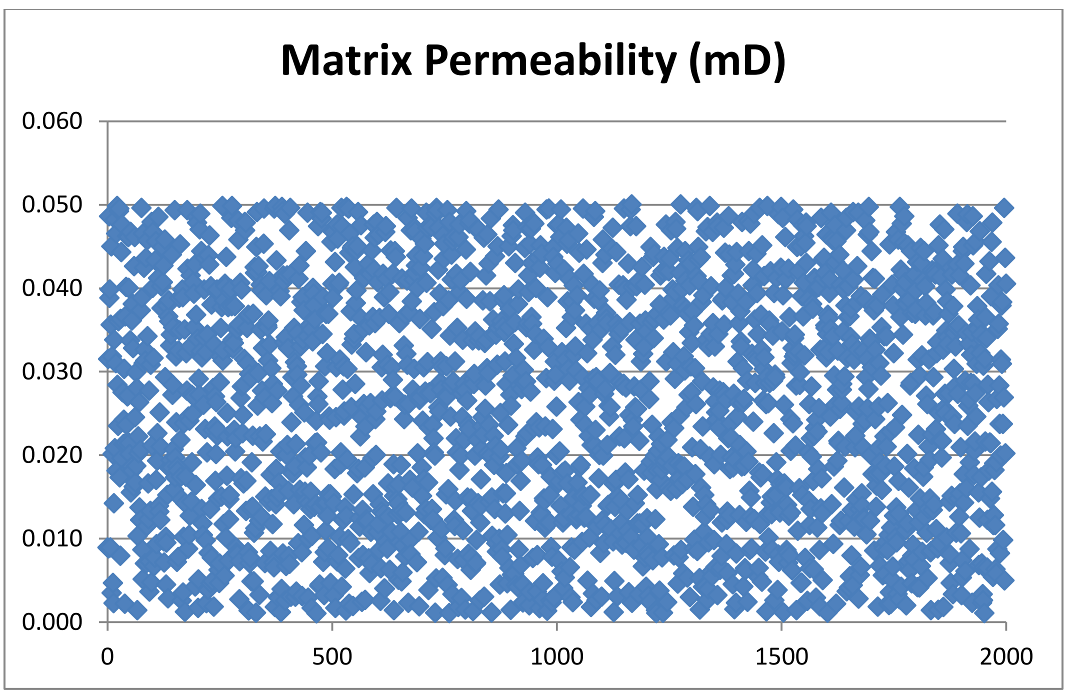


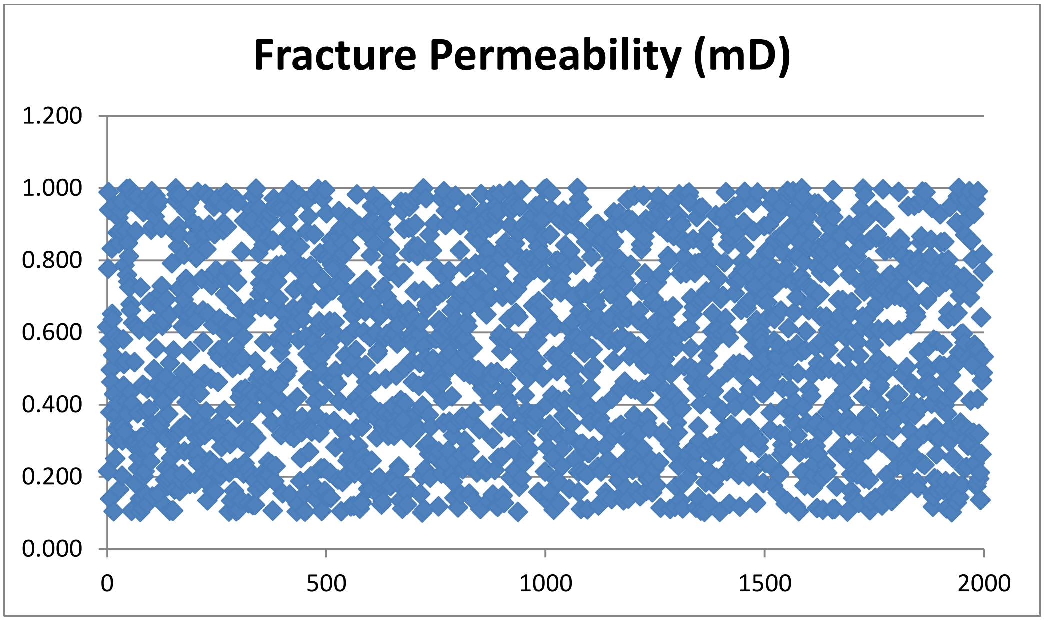


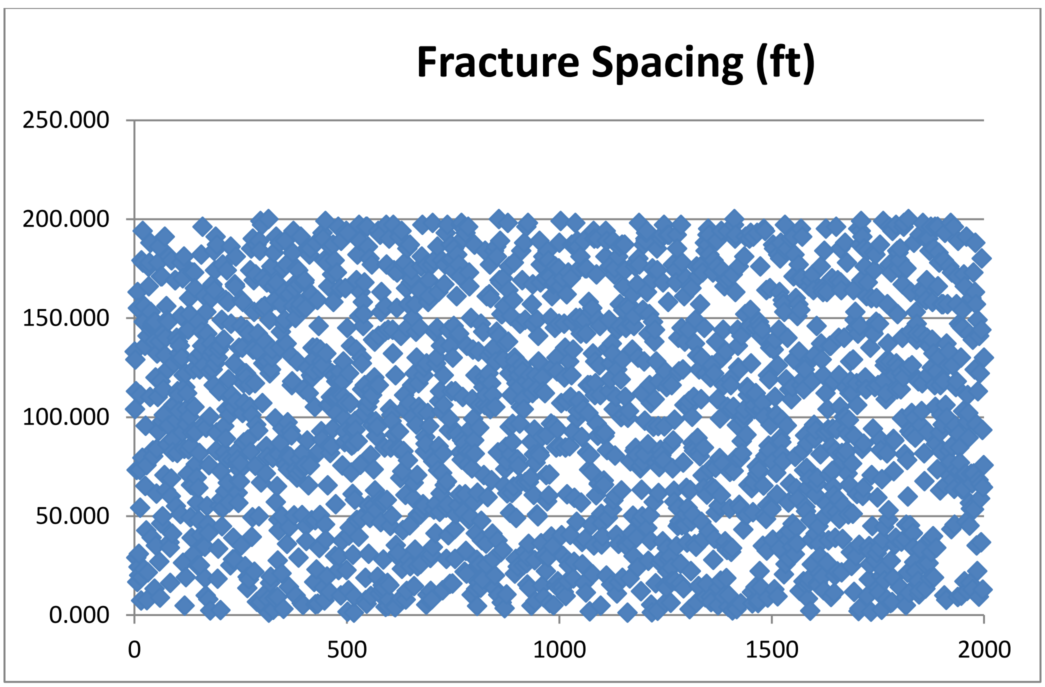

Thickness (ft)
450.00
400.00
350.00
300.00
250.00
200.00
150.00
100.00
50.00
0
200
400
600
800
1000
1200
1400
1600
1800
2000

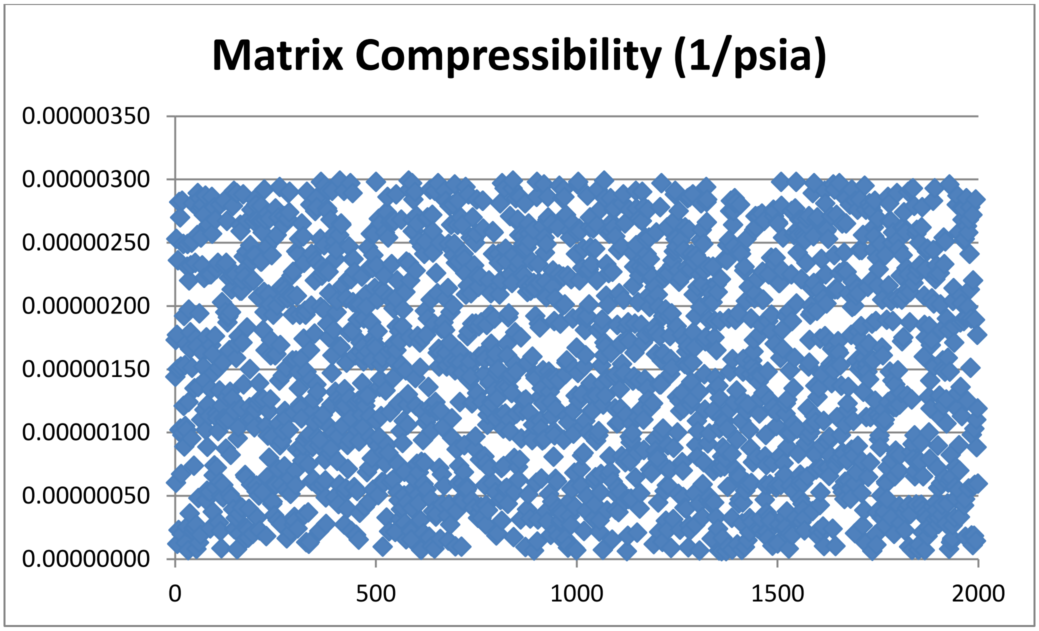

Matrix Porosity
0.450
0.400
0.350
0.300
0.250
0.200
0.150
0.100
0.050
0.000
0
200
400
600
800
1000
1200
1400
1600
1800
2000

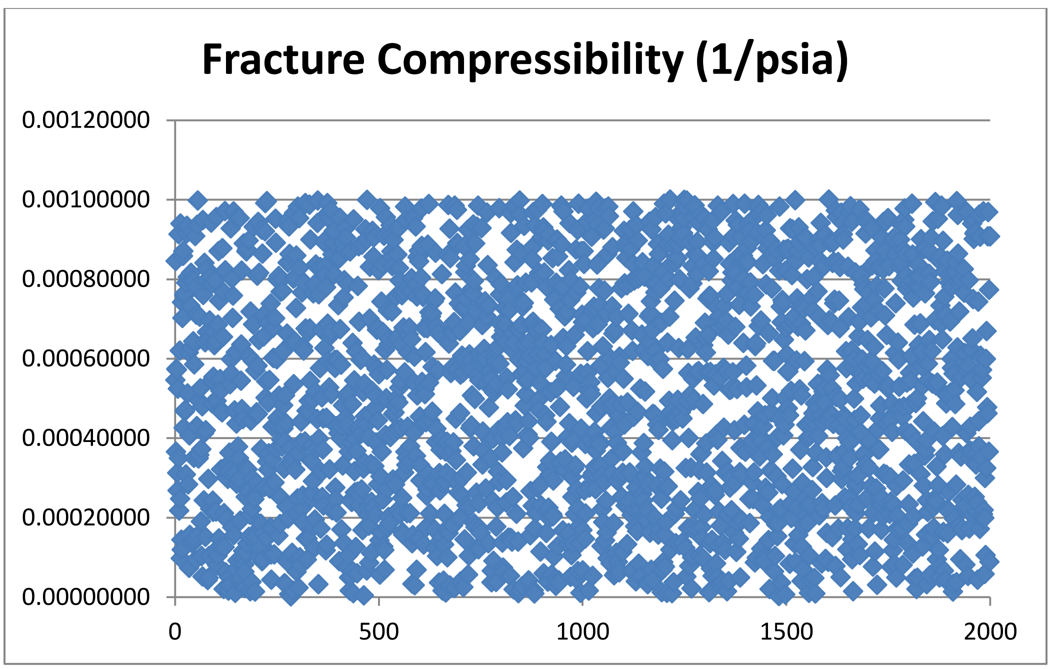

Fracture Porosity
0.035
0.030
0.025
0.020
0.015
0.010
0.005
0.000
0
500
1000
1500
2000

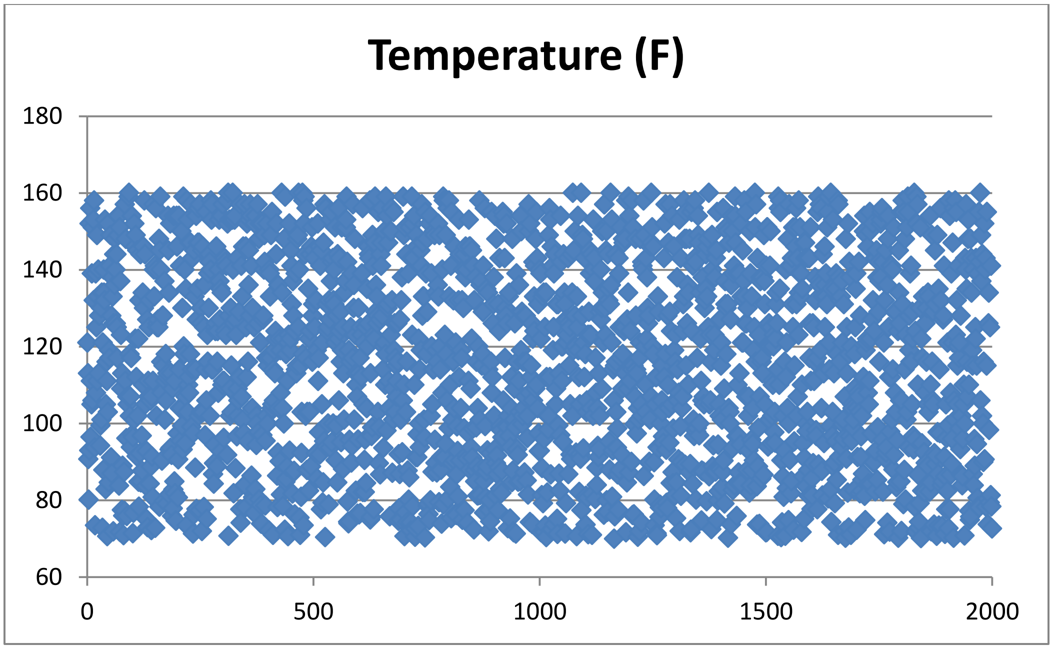

Matrix Permeability (mD)
0.060
0.050
0.040
0.030
0.020
0.010
0.000
0
500
1000
1500
2000


Fracture Permeability (mD)
1.200
1.000
0.800
0.600
0.400
0.200
0.000
0
500
1000
1500
2000

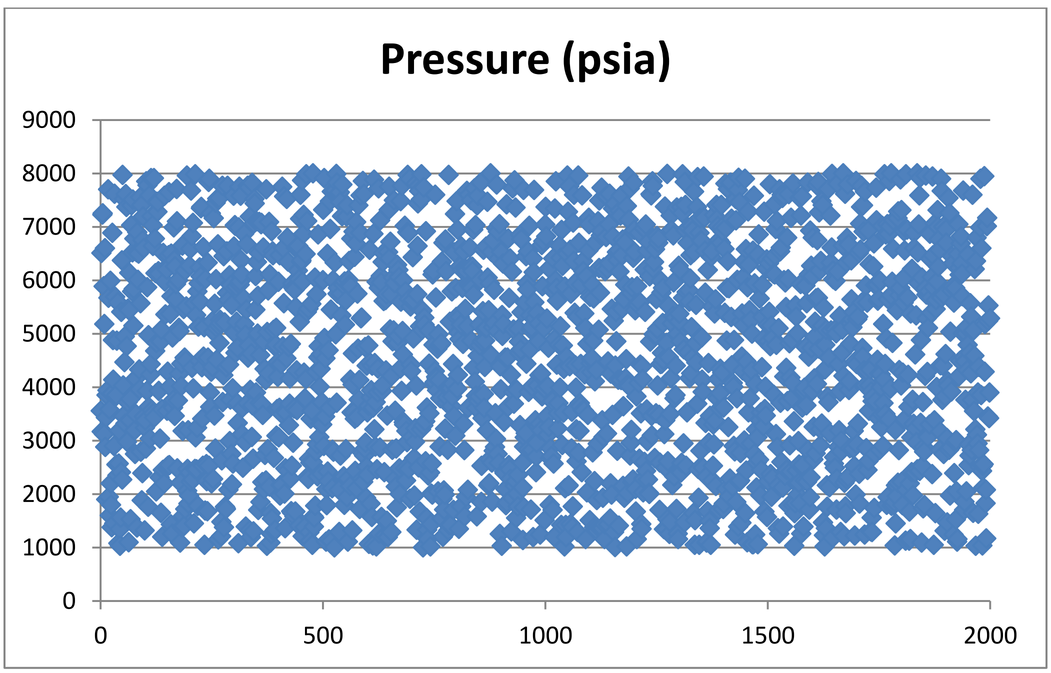

Fracture Spacing (ft)
250.000
200.000
150.000
100.000
50.000
0.000
0
500
1000
1500
2000


Matrix Compressibility (1/psia)
0.00000350
0.00000300
0.00000250
0.00000200
0.00000150
0.00000100
0.00000050
0.00000000
0
500
1000
1500
2000

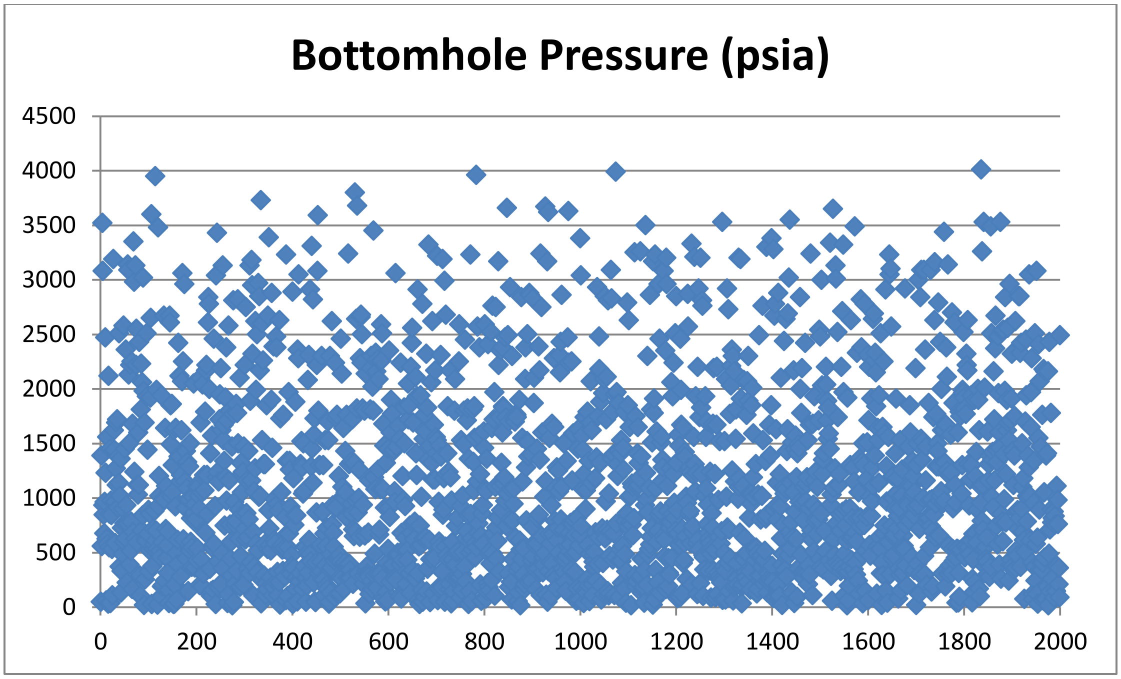

Fracture Compressibility (1/psia)
0.00120000
0.00100000
0.00080000
0.00060000
0.00040000
0.00020000
0.00000000
0
500
1000
1500
2000

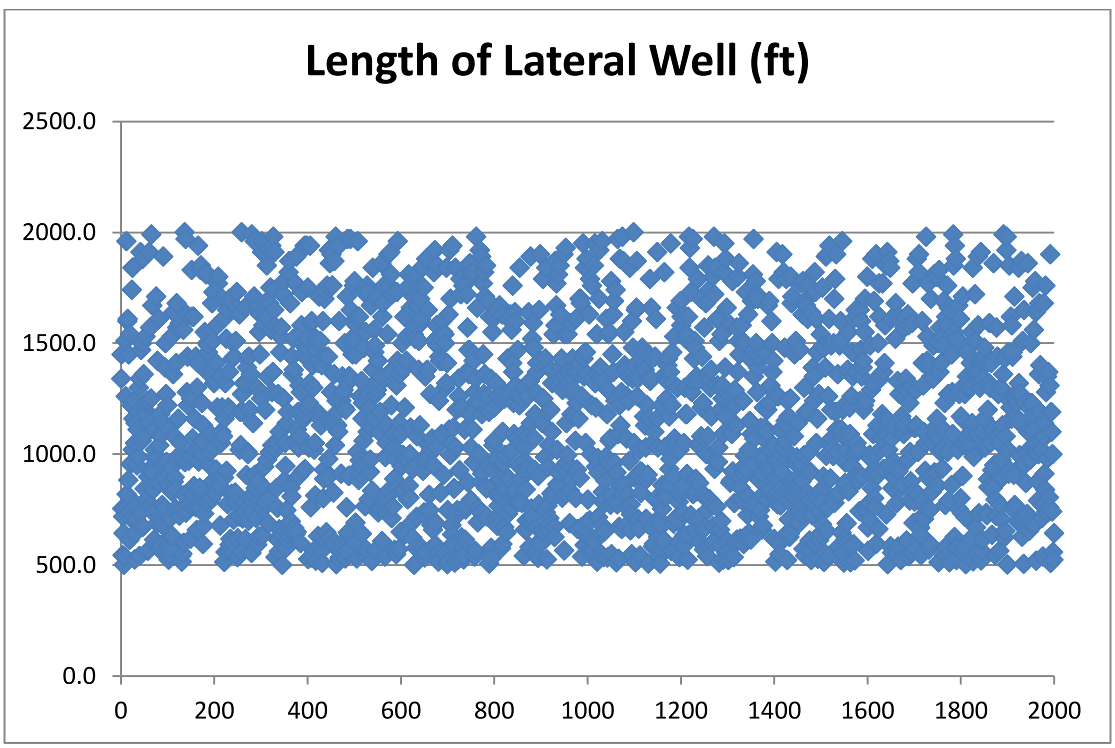


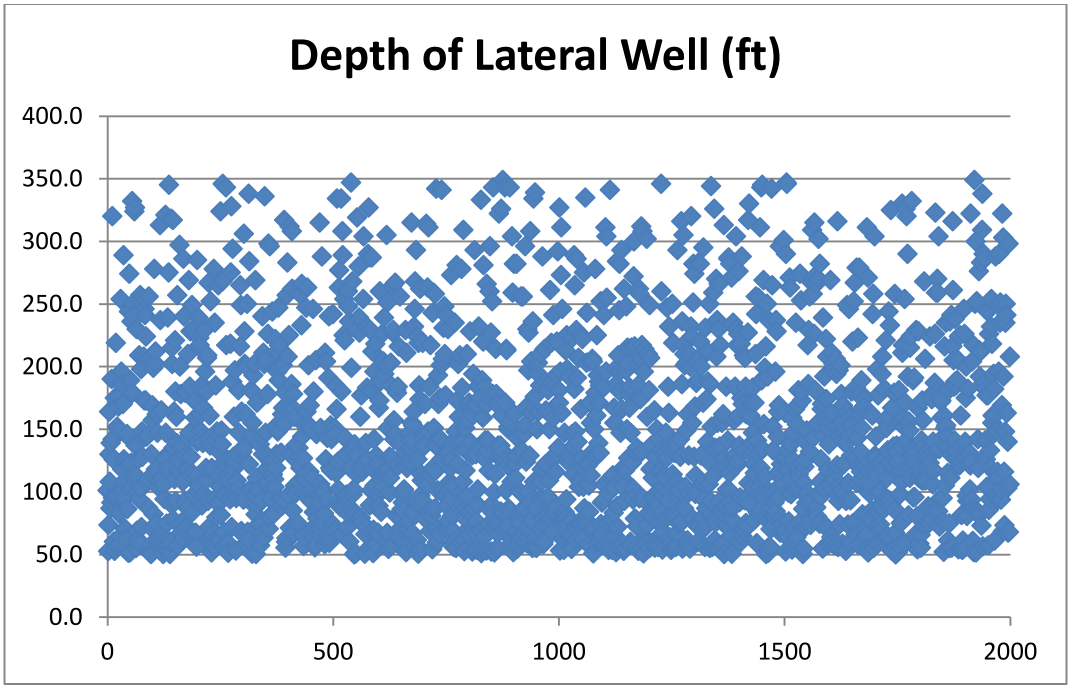
Temperature (F)
180
160
140
120
100
80
60
0
500
1000
1500
2000



Pressure (psia)
9000
8000
7000
6000
5000
4000
3000
2000
1000
0
0
500
1000
1500
2000


Bottomhole Pressure (psia)
4500
4000
3500
3000
2500
2000
1500
1000
500
0
0
200
400
600
800
1000
1200
1400
1600
1800
2000


Length of Lateral Well (ft)
2500.0
2000.0
1500.0
1000.0
500.0
0.0
0
200
400
600
800
1000
1200
1400
1600
1800
2000



Depth of Lateral Well (ft)
400.0
350.0
300.0
250.0
200.0
150.0
100.0
50.0
0.0
0
500
1000
1500
2000

Appendix E: Table of Parameter Sensitivity Check

Fracture Permeability	Original	5% Difference		10% Difference	
TIME	Gas Rate SC - Monthly	Gas Rate SC - Monthly	Deviation	Gas Rate SC - Monthly	Deviation
(day)	(ft3/day)	(ft3/day)		(ft3/day)	
31	358550176	373268384	3.78%	387942848	7.58%
120	258433568	269809536	4.00%	281056064	8.05%
212	240374688	250980160	4.03%	261514272	8.08%
304	230018944	240182304	4.04%	250283568	8.10%
396	222854368	232689792	4.03%	242460480	8.09%
486	217433152	226995680	4.01%	236489936	8.06%
670	208886480	217937856	3.95%	226902544	7.94%
762	205293632	214096592	3.91%	222801792	7.86%
943	198925008	207255488	3.81%	215463344	7.68%
1035	195930976	204023616	3.75%	211982528	7.57%
1127	193050032	200908528	3.70%	208627520	7.47%
1216	190348224	197984800	3.64%	205473936	7.36%
1308	187641200	195058384	3.59%	202322320	7.26%
1492	182442704	189437088	3.48%	196267632	7.04%
1673	177575072	184172016	3.37%	190596656	6.83%
1765	175180896	181586512	3.32%	187818784	6.73%
1857	172840624	179061264	3.26%	185101152	6.62%
1947	170593488	176635904	3.21%	182497008	6.52%
2039	168353520	174220064	3.16%	179904784	6.42%
2131	166152784	171852304	3.11%	177364720	6.32%
2312	161953648	167337440	3.01%	172528976	6.13%
2496	157862480	162942576	2.91%	167826064	5.94%
2588	155875264	160810544	2.86%	165549216	5.84%
2677	153985568	158782800	2.81%	163381248	5.75%
2769	152076288	156735680	2.76%	161188960	5.65%
2861	150201120	154719920	2.71%	159036848	5.56%
2953	148356592	152741168	2.67%	156929120	5.46%
3042	146599664	150859504	2.62%	154924912	5.37%
3134	144819760	148959104	2.58%	152897600	5.28%
3226	143071424	147088160	2.53%	150912496	5.20%
3408	139698720	143495440	2.45%	147093360	5.03%
3500	138044192	141733232	2.41%	145227776	4.95%
3592	136418464	140001808	2.36%	143392768	4.86%
3653	135359328	138870784	2.34%	142194096	4.81%

Fracture Porosity	Original	5% Difference		10% Difference	
TIME	Gas Rate SC - Monthly	Gas Rate SC - Monthly	Deviation	Gas Rate SC - Monthly	Deviatio n
(day)	(ft3/day)	(ft3/day)		(ft3/day)	
31	358550176	359353536	0.22%	360155680	0.45%
151	250881184	251260128	0.15%	251630752	0.30%
273	233020752	233362912	0.15%	233700016	0.29%
396	222854368	223183120	0.15%	223506560	0.29%
517	215813488	216142752	0.15%	216465136	0.30%
639	210169584	210513104	0.16%	210851152	0.32%
762	205293632	205657248	0.18%	206012416	0.35%
882	200988336	201376272	0.19%	201754688	0.38%
1004	196922032	197336544	0.21%	197742320	0.42%
1127	193050032	193493456	0.23%	193927888	0.45%
1247	189431280	189900608	0.25%	190362000	0.49%
1369	185883040	186381104	0.27%	186866080	0.53%
1492	182442704	182963824	0.29%	183474368	0.57%
1612	179190224	179732480	0.30%	180266288	0.60%
1734	175978352	176544160	0.32%	177100112	0.64%
1857	172840624	173425168	0.34%	174002304	0.67%
1978	169836544	170441168	0.36%	171035216	0.71%
2100	166885232	167506736	0.37%	168120592	0.74%
2223	163998560	164639120	0.39%	165264880	0.77%
2343	161256608	161907744	0.40%	162549360	0.80%
2465	158539232	159203328	0.42%	159857600	0.83%
2588	155875264	156551760	0.43%	157219808	0.86%
2708	153341552	154029680	0.45%	154707248	0.89%
2830	150826736	151524688	0.46%	152211728	0.92%
2953	148356592	149064800	0.48%	149762448	0.95%
3073	145999408	146716080	0.49%	147422176	0.97%
3195	143652720	144380176	0.51%	145094512	1.00%
3318	141354816	142086272	0.52%	142807168	1.03%
3439	139141392	139877072	0.53%	140602208	1.05%
3469	138597712	139333904	0.53%	140065728	1.06%
3500	138044192	138786224	0.54%	139515568	1.07%
3531	137492416	138232336	0.54%	138968032	1.07%
3561	136961504	137704240	0.54%	138433872	1.08%
3592	136418464	137162624	0.55%	137896208	1.08%
3622	135891088	136633824	0.55%	137370016	1.09%
3653	135359328	136101360	0.55%	136834944	1.09%

Matrix Permeability	Original	5% Difference		10% Difference	
TIME	Gas Rate SC - Monthly	Gas Rate SC - Monthly	Deviation	Gas Rate SC - Monthly	Deviation
(day)	(ft3/day)	(ft3/day)		(ft3/day)	
31	358550176	358558720	0.00%	358579072	0.01%
151	250881184	250875248	0.00%	250872080	0.00%
273	233020752	233018576	0.00%	233018032	0.00%
396	222854368	222853312	0.00%	222853584	0.00%
517	215813488	215813216	0.00%	215813744	0.00%
639	210169584	210169584	0.00%	210169584	0.00%
762	205293632	205294160	0.00%	205294688	0.00%
882	200988336	200988336	0.00%	200989392	0.00%
1004	196922032	196922576	0.00%	196923120	0.00%
1127	193050032	193050560	0.00%	193051088	0.00%
1247	189431280	189431808	0.00%	189432336	0.00%
1369	185883040	185884128	0.00%	185885216	0.00%
1492	182442704	182443760	0.00%	182444832	0.00%
1612	179190224	179191280	0.00%	179192336	0.00%
1734	175978352	175979456	0.00%	175980544	0.00%
1857	172840624	172839568	0.00%	172842736	0.00%
1978	169836544	169837600	0.00%	169837600	0.00%
2100	166885232	166886336	0.00%	166887424	0.00%
2223	163998560	164000672	0.00%	164001728	0.00%
2343	161256608	161256608	0.00%	161257664	0.00%
2465	158539232	158540320	0.00%	158540320	0.00%
2588	155875264	155877376	0.00%	155876320	0.00%
2708	153341552	153341552	0.00%	153343664	0.00%
2830	150826736	150827824	0.00%	150827824	0.00%
2953	148356592	148358704	0.00%	148358704	0.00%
3073	145999408	145997296	0.00%	145999408	0.00%
3195	143652720	143654912	0.00%	143654912	0.00%
3318	141354816	141354816	0.00%	141352688	0.00%
3439	139141392	139141392	0.00%	139141392	0.00%
3531	137492416	137492416	0.00%	137494528	0.00%
3561	136961504	136961504	0.00%	136961504	0.00%
3592	136418464	136420576	0.00%	136420576	0.00%
3622	135891088	135893264	0.00%	135893264	0.00%
3653	135359328	135355088	0.00%	135357216	0.00%

Matrix Porosity	Original	5% Difference		10% Difference	
TIME	Gas Rate SC - Monthly	Gas Rate SC - Monthly	Deviation	Gas Rate SC - Monthly	Deviation
(day)	(ft3/day)	(ft3/day)		(ft3/day)	
31	358550176	360858208	0.64%	362983648	1.24%
151	250881184	251952224	0.43%	253009520	0.85%
273	233020752	233980320	0.41%	234920480	0.82%
396	222854368	223772144	0.41%	224666400	0.81%
517	215813488	216728880	0.42%	217617840	0.84%
639	210169584	211121488	0.45%	212031344	0.89%
762	205293632	206294112	0.49%	207248608	0.95%
882	200988336	202054352	0.53%	203072816	1.04%
1004	196922032	198063984	0.58%	199146432	1.13%
1127	193050032	194274080	0.63%	195429408	1.23%
1247	189431280	190733536	0.69%	191963408	1.34%
1369	185883040	187263664	0.74%	188568912	1.44%
1492	182442704	183898240	0.80%	185272384	1.55%
1612	179190224	180715520	0.85%	182157312	1.66%
1734	175978352	177575248	0.91%	179083680	1.76%
1857	172840624	174503344	0.96%	176077264	1.87%
1978	169836544	171563728	1.02%	173195792	1.98%
2100	166885232	168673280	1.07%	170364112	2.08%
2223	163998560	165843072	1.12%	167588240	2.19%
2343	161256608	163150816	1.17%	164946720	2.29%
2465	158539232	160482368	1.23%	162326112	2.39%
2588	155875264	157867760	1.28%	159756688	2.49%
2708	153341552	155377392	1.33%	157307536	2.59%
2830	150826736	152903136	1.38%	154875760	2.68%
2953	148356592	150479120	1.43%	152487472	2.78%
3073	145999408	148159984	1.48%	150210624	2.88%
3195	143652720	145852560	1.53%	147940960	2.99%
3318	141354816	143585152	1.58%	145707664	3.08%
3439	139141392	141403440	1.63%	143555552	3.17%
3561	136961504	139250896	1.67%	141435424	3.27%
3592	136418464	138718576	1.69%	140904512	3.29%
3622	135891088	138195760	1.70%	140389040	3.31%
3653	135359328	137667888	1.71%	139868624	3.33%

Thickness	Original	5% Difference		10% Difference	
TIME	Gas Rate SC - Monthly	Gas Rate SC - Monthly	Deviation	Gas Rate SC - Monthly	Deviation
(day)	(ft3/day)	(ft3/day)		(ft3/day)	
31	358550176	373709760	4.23%	388854560	8.45%
151	250881184	260589760	3.87%	270124736	7.67%
273	233020752	242235536	3.95%	251324272	7.85%
396	222854368	231782864	4.01%	240599840	7.96%
517	215813488	224539808	4.04%	233156208	8.04%
639	210169584	218719296	4.07%	227159792	8.08%
762	205293632	213686992	4.09%	221975712	8.13%
882	200988336	209246928	4.11%	217405104	8.17%
1004	196922032	205054496	4.13%	213085936	8.21%
1127	193050032	201055984	4.15%	208969456	8.25%
1247	189431280	197321504	4.17%	205118160	8.28%
1369	185883040	193659968	4.18%	201345168	8.32%
1492	182442704	190103024	4.20%	197678768	8.35%
1612	179190224	186741664	4.21%	194211712	8.38%
1734	175978352	183423248	4.23%	190789488	8.42%
1857	172840624	180178544	4.25%	187443536	8.45%
1978	169836544	177076160	4.26%	184240720	8.48%
2100	166885232	174028656	4.28%	181096720	8.52%
2223	163998560	171042624	4.30%	178016912	8.55%
2343	161256608	168204480	4.31%	175085760	8.58%
2465	158539232	165393200	4.32%	172182736	8.61%
2588	155875264	162640272	4.34%	169338688	8.64%
2708	153341552	160015664	4.35%	166628448	8.66%
2830	150826736	157408736	4.36%	163933936	8.69%
2953	148356592	154855232	4.38%	161290432	8.72%
3073	145999408	152411360	4.39%	158766240	8.74%
3195	143652720	149981328	4.41%	156257488	8.77%
3318	141354816	147597648	4.42%	153787616	8.80%
3439	139141392	145306000	4.43%	151413536	8.82%
3561	136961504	143045424	4.44%	149079104	8.85%
3592	136418464	142481600	4.44%	148496112	8.85%
3622	135891088	141937872	4.45%	147932224	8.86%
3653	135359328	141382288	4.45%	147356640	8.86%

Fracture Spacing	Original	5% Difference		10% Difference		30% Difference		60% Difference		120% Difference	
TIME	Gas Rate SC - Monthly	Gas Rate SC - Monthly	Deviation	Gas Rate SC - Monthly	Deviation	Gas Rate SC - Monthly	Deviation	Gas Rate SC - Monthly	Deviation	Gas Rate SC - Monthly	Deviation
(day)	(ft3/day)	(ft3/day)		(ft3/day)		(ft3/day)		(ft3/day)		(ft3/day)	
31	358550176	358532384	0.00%	358299872	0.07%	357241600	0.36%	356062912	0.69%	352167712	1.78%
181	245147504	245155296	0.00%	245158160	0.00%	245186416	0.02%	245031184	0.05%	245112416	0.01%
304	230018944	230022368	0.00%	230023968	0.00%	230034800	0.01%	229995680	0.01%	230023968	0.00%
425	220962256	220963376	0.00%	220962816	0.00%	220967056	0.00%	220951232	0.00%	220958576	0.00%
547	214322656	214322928	0.00%	214322112	0.00%	214321008	0.00%	214311456	0.01%	214301632	0.01%
670	208886480	208885952	0.00%	208884368	0.00%	208880672	0.00%	208871696	0.01%	208848432	0.02%
790	204240016	204240016	0.00%	204237680	0.00%	204231824	0.00%	204222464	0.01%	204190288	0.02%
912	199960720	199959616	0.00%	199957984	0.00%	199950336	0.01%	199940512	0.01%	199903376	0.03%
1035	195930976	195930448	0.00%	195927264	0.00%	195919872	0.01%	195908768	0.01%	195868608	0.03%
1155	192180224	192179056	0.00%	192177296	0.00%	192168528	0.01%	192156816	0.01%	192114096	0.03%
1277	188543248	188542160	0.00%	188539968	0.00%	188531232	0.01%	188520864	0.01%	188476080	0.04%
1400	185008128	185006016	0.00%	185003904	0.00%	184994384	0.01%	184982752	0.01%	184939424	0.04%
1520	181662288	181661104	0.00%	181658768	0.00%	181650576	0.01%	181638880	0.01%	181594400	0.04%
1642	178388992	178386800	0.00%	178385712	0.00%	178376976	0.01%	178364960	0.01%	178322368	0.04%
1765	175180896	175178784	0.00%	175177728	0.00%	175168208	0.01%	175158704	0.01%	175115360	0.04%
1886	172104320	172103184	0.00%	172102048	0.00%	172094144	0.01%	172081712	0.01%	172041040	0.04%
2008	169101456	169099264	0.00%	169095984	0.00%	169089440	0.01%	169079600	0.01%	169038096	0.04%
2131	166152784	166151728	0.00%	166149616	0.00%	166142224	0.01%	166131648	0.01%	166091472	0.04%
2251	163343792	163341456	0.00%	163337952	0.00%	163332096	0.01%	163321568	0.01%	163281776	0.04%
2373	160578496	160577392	0.00%	160574128	0.00%	160568656	0.01%	160557744	0.01%	160519504	0.04%
2496	157862480	157862480	0.00%	157860368	0.00%	157852976	0.01%	157843456	0.01%	157806464	0.04%
2616	155268832	155267664	0.00%	155264144	0.00%	155258288	0.01%	155250096	0.01%	155213824	0.04%
2738	152715264	152714176	0.00%	152710896	0.00%	152704336	0.01%	152695600	0.01%	152661744	0.04%
2861	150201120	150201120	0.00%	150198992	0.00%	150191600	0.01%	150183136	0.01%	150149312	0.03%
2981	147790704	147790704	0.00%	147790704	0.00%	147783680	0.00%	147774320	0.01%	147741552	0.03%
3103	145413456	145413456	0.00%	145411280	0.00%	145404720	0.01%	145395984	0.01%	145363216	0.03%
3226	143071424	143069312	0.00%	143067200	0.00%	143060864	0.01%	143054512	0.01%	143022800	0.03%
3347	140809744	140809744	0.00%	140807488	0.00%	140800704	0.01%	140793920	0.01%	140764544	0.03%
3469	138597712	138595536	0.00%	138595536	0.00%	138588976	0.01%	138582432	0.01%	138551840	0.03%
3500	138044192	138042064	0.00%	138042064	0.00%	138037840	0.00%	138029392	0.01%	137999792	0.03%
3531	137492416	137494528	0.00%	137490304	0.00%	137483952	0.01%	137477616	0.01%	137450128	0.03%
3622	135891088	135891088	0.00%	135886704	0.00%	135882336	0.01%	135875792	0.01%	135849568	0.03%
3653	135359328	135357216	0.00%	135357216	0.00%	135348752	0.01%	135342416	0.01%	135314928	0.03%