The Pennsylvania State University

The Graduate School

John and Willie Leone Family Department of Energy and Mineral Engineering

PRESSURE TRANSIENT ANALYSIS OF SHALE GAS RESERVOIRS WITH HORIZONTAL BOREHOLES: AN ARTIFICIAL INTELLIGENCE BASED APPROACH

A Thesis in

Energy and Mineral Engineering

by

Xuwei Zhong

Submitted in Partial Fulfillment
of the Requirements
for the Degree of

Master of Science

August 2018

The thesis of Xuwei Zhong was reviewed and approved* by the following:

Turgay Ertekin
Professor Emeritus of Petroleum and Natural Gas Engineering
George E. Trimble Chair in Earth and Mineral Sciences
Thesis Advisor

Luis F. Ayala H.
William A. Fustos Family Professor of Petroleum and Natural Gas Engineering
Associate Department Head for Graduate Education

Hamid Emami-Meybodi
Assistant Professor of Petroleum and Natural Gas Engineering

*Signatures are on file in the Graduate School

ABSTRACT

In common oil and gas industry, obtaining reservoir properties is critical to field development. Well testing has been one of the important tools to estimate reservoir characteristics such as permeability and reservoir thickness. The purpose of this study is to develop an expert system that can estimate reservoir characteristics based on constant flow rate pressure transient data from horizontal wells located in shale gas reservoirs.

A commercial compositional reservoir simulator CMG-GEM was used in this study. An isotropic square reservoir model with one lateral well at the center was built. The model utilized stimulated reservoir volume (SRV) approach to represent the fractured zone of the reservoir. The SRV zone was more prolific than the rest of the reservoir due to its higher permeability and smaller fracture spacing. 37 x 37 reservoir blocks configuration was selected to use in the model after performing a grid block sensitivity analysis.

Since this study requires training of the expert system, large number of data need to be generated. 15 reservoir and well completion parameters were selected as variables in specified ranges. Several MATLAB scripts were created to randomize the input variables and to build thousands of simulation models. The pressure transient data were collected in a numerical table after each model ran.

The expert systems included seven Artificial Neural Networks (ANN). The ANNs were all generated by using MATLAB ANN Toolbox. The ANNs were classified into forward ANN and inverse ANN. The forward ANN was able to predict pressure transient data, while the inverse ANN was capable of predicting reservoir properties.

During the final stage of this study, five graphical user interfaces (GUI) were developed. The GUIs allow the user to input required parameters, and to view results in numerical and graphical formats.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

ACKNOWLEDGEMENTS

I would like to thank my graduate and undergraduate advisor Dr. Turgay Ertekin for his continuous help and valuable support. He guided me through all the challenges and frustrations I encountered, especially for the completion of this thesis. I appreciate the opportunity to work with one of the finest professors at The Pennsylvania State University. I also want to extend my appreciation to Dr. Luis F. Ayala H. and Dr. Hamid Emami-Meybodi for their interests in serving on my committee.

I would like to thank all the wonderful friends I met at Penn State for bringing happiness and support to my life. I sincerely thank my fellow graduate student Dr. Jian Zhang for assisting me in every stage of this master thesis. His expertise in Reservoir Simulation and Artificial Neural Network helped me to overcome obstacles and to optimize my results.

And lastly, I want to thank my family for their continuous support and encouragement. I am deeply grateful for my father Yuhui Zhong and my mother Xiaohong Peng for raising me up and educating me throughout my life.

Chapter 1

Introduction

With the increasing consumption in oil and gas, hydrocarbon productions from conventional reservoirs can no longer satisfy the demand. In the early 21st century, a boom in shale natural gas production occurred. According to the U.S. Energy Information Administration website, the U.S. dry natural gas production was equal to around 97% of the U.S. natural gas consumption (U.S. Energy Information Administration, 2017). Thus, the understandings of shale gas reservoirs have become the priority of many research centers and universities. As shown in Figure 1-1, the production of shale gas is projected to increase, and has already surpassed other natural gas plays.

Drilling and completion tasks have been reported to be more difficult when conducted in shale gas reservoirs compared to conventional reservoirs. Shale reservoirs have remarkably lower permeability and porosity characteristics in comparison with traditional sandstone reservoirs. In order to increase the areal exposure of wellbore to the reservoir rock, lateral wells and hydraulic fracturing are commonly used by oil and gas companies.

Well testing is a critical application at early stages of a field development. Commonly, the well is put on production for a short period of time before being shut in. The data collected during well testing period can help experienced petroleum engineers to estimate reservoir properties. However, the accuracy of this approach is limited by human errors. Engineers with various backgrounds may stand for their own analyses.

Artificial neural network (ANN) is an effective decision-making tool based on machine learning. ANN can provide an effective mapping of describing the existing complex relationships

if it is structured and trained correctly. ANN requires a large number of data sets in training before it can predict accurately. However, it is not economic to run thousands of real field well tests, and this approach defeats the original purpose of well testing. Alternately, simulation model can be built using similar configuration of the actual reservoir. The simulation model can produce required number of data in short period and it is cost efficient.

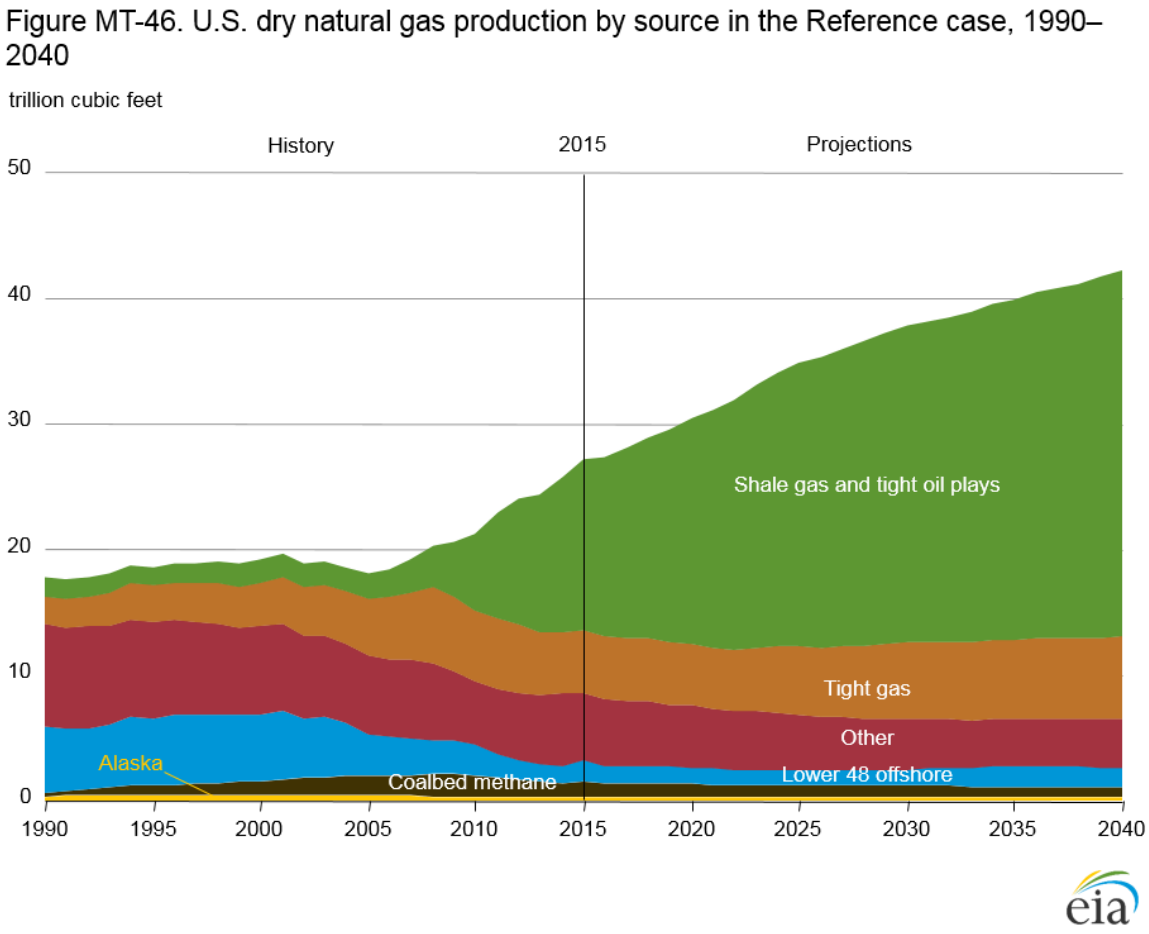


Figure 1-1. U.S. Dry Natural Gas Production and Projection of Recent Years (U.S. EIA, 2017)

Chapter 2

Literature Review

2.1 Reservoir Characterization

In oil and gas industry, reservoirs are classified based on their property related characteristics. All reservoirs fall into either conventional or unconventional category. Conventional reservoirs are the first priority of companies due to their simplicity. This type of reservoirs has been developed since 1859 when the first oil well was drilled and put into production in Pennsylvania. Conventional reservoirs contain hydrocarbons in porous reservoir rocks. The extraction is usually straightforward and economic. In contrast, unconventional reservoirs such as tight gas, shale gas and coalbed methane reservoirs require higher technical expertise (Belvalkar & Oyewole, 2010). Advanced drilling & completion technologies and high oil price are essential to make the development of unconventional reservoirs profitable. Typical vertical wells without stimulation would make it difficult to achieve economical production from an unconventional well (Belvalkar & Oyewole, 2010).

2.1.1 Shale Reservoir

Shale is a fine-grained, fissile, detrital sedimentary rock. It is formed over time as clay and silt-sized particles deposit and consolidate. It has ultra-low permeability which makes it good seal rock to many conventional reservoirs (Belvalkar & Oyewole, 2010). It has permeability lower than 0.001 mD, which makes shale difficult to develop. However, shale is organic rich with total organic

carbon (TOC) ranging from 1% to 10%. Shale is more radioactive than other sedimentary rocks, and its gamma-ray measurement is usually higher than 140 API (Alqahtani & Ertekin, 2017). A sample of shale rock is shown in Figure 2-1.

Figure 2-1. Barnett Shale Sample with Natural Fractures (Walton & McLennan, 2013)

Shale serves as both source and reservoir rock in shale gas reservoirs. The amount of hydrocarbon gas stored in the shale reservoir is strongly affected by the total organic carbon and the adsorption ability of methane on the internal surface of solid (Yu, Sepehrnoori & Patzek, 2014). Because shale has ultra-tight structure, the development of shale gas is challenging with traditional methods used in conventional reservoirs. The commercial exploitation of shale gas reservoirs has been improved with multi-stage hydraulic fracturing of long horizontal wells. In order to maximize the gas production, large volumes of slickwater are usually injected at high pressures and rates to create and propagate extensive hydraulic fracture systems (Walton & McLennan, 2013).

2.1.2 Fractured Reservoirs and Double-Porosity Model

Natural and man-made fractures commonly exist in petroleum reservoirs. Fractures are important channels for fluid flow. Natural fractures are the result of interactions between tectonic stresses. Artificial fractures are the results of hydraulic fracking. Fracking is essential in shale gas plays due to the low permeability and low porosity property of shale.

To better simulate real shale gas reservoir, the double porosity model is implemented in CMG-GEM software. The paper published by Warren and Root in 1963 questions the accuracy of the "two parameters" approach when solving single-phase flow properties. They state that absolute permeability and the effective porosity are not sufficient to represent a reservoir. A model that consists of internal voids of reservoir rocks is suggested as illustrated by Figure 2-2. The model has three basic assumptions according to the paper:

1) The material inside of the matrix is homogenous and isotropic.
2) The fracture porosity is represented by an orthogonal and continuous system; The fractures have to be normal to each of the principal axes and are uniformly spaced.
3) Flow can occur between matrix and fracture element, but not between matrixes. (Warren & Root, 1962)

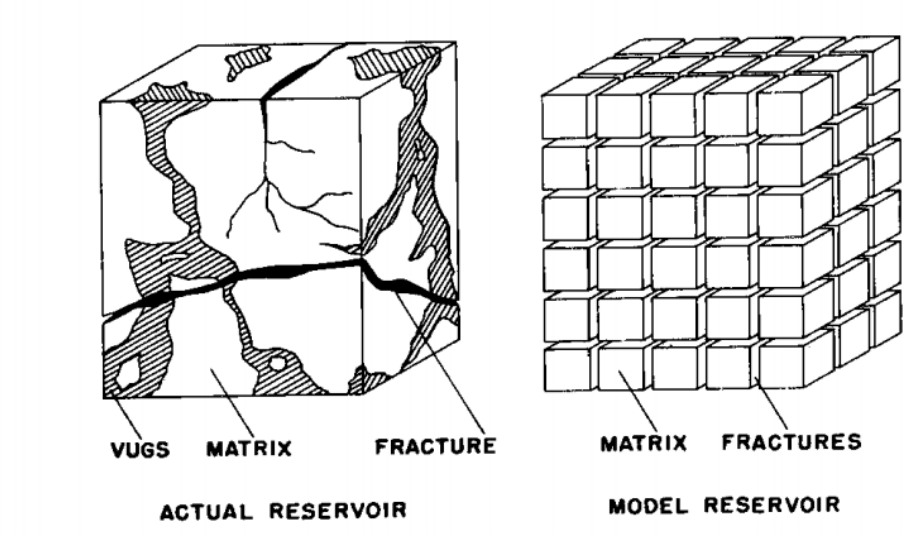


Figure 2-2. Warren & Root Double Porosity Model (Warren & Root, 1963)

Since no interactions are allowed between matrix elements directly, the main channel for the fluid flow is the interconnected fractures. Thus, the transport characteristics of fractures are crucial in determining the productivity of the field. The hydraulic fracturing increases the volume of fractures and leads to better connectivity around the wellbore. Unconventional reservoirs become profitable thanks to the development of technology.

2.1.3 Adsorbed Gas

There are five mechanisms for the existence of methane in shale reservoirs: (1) adsorption on internal surface area; (2) conventional storage in natural and hydraulic fractures; (3) conventional storage in matrix porosity; (4) solution in formation water; (5) adsorption in organic matter (Clarkson & Haghshenas, 2013). The adsorption process in shale reservoirs is mostly physical adsorption controlled by weak van der Waals force. Thus, the adsorption is fully reversible and allows gas to desorb from the solid (Sapag, Vallone, Garca & Solar, 2010). Since a major

portion of gas in shale reservoir is adsorbed to solid, investigation of gas adsorption can provide important insights into evaluation of shale properties and well performance (Yu et al., 2014).

The most commonly used adsorption model for shale reservoirs is the Langmuir isotherm (Langmuir, 1918). The model assumes dynamic equilibrium at constant temperature and pressure between adsorbed and non-adsorbed gas. The classic model also assumes that there is only a single layer of molecules adsorbed to the surface as illustrated by Figure 2-3.

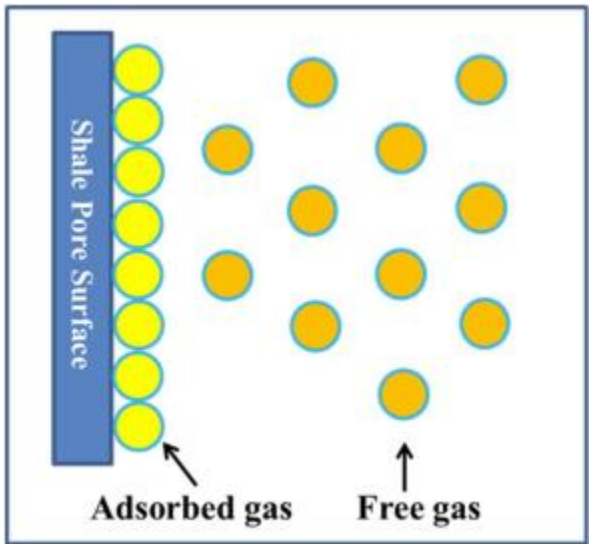


Figure 2-3. Monolayer Langmuir Adsorption (Yu et al., 2014)

The formulation of Langmuir isotherm is expressed as follows:

$$v(p) = \frac{v_L p}{p + p_L} \tag{1}$$

where $v(p)$ is the adsorbed gas volume as a function of pressure. v_L is the Langmuir volume which is the maximum amount of gas that can be adsorbed to shale at infinite pressure. Langmuir pressure p_L is the pressure at which one half of the Langmuir volume can be adsorbed. Figure 2-4 shows an example of Langmuir isotherm. The adsorption option is embedded in the CMG-GEM software and is used in this study.

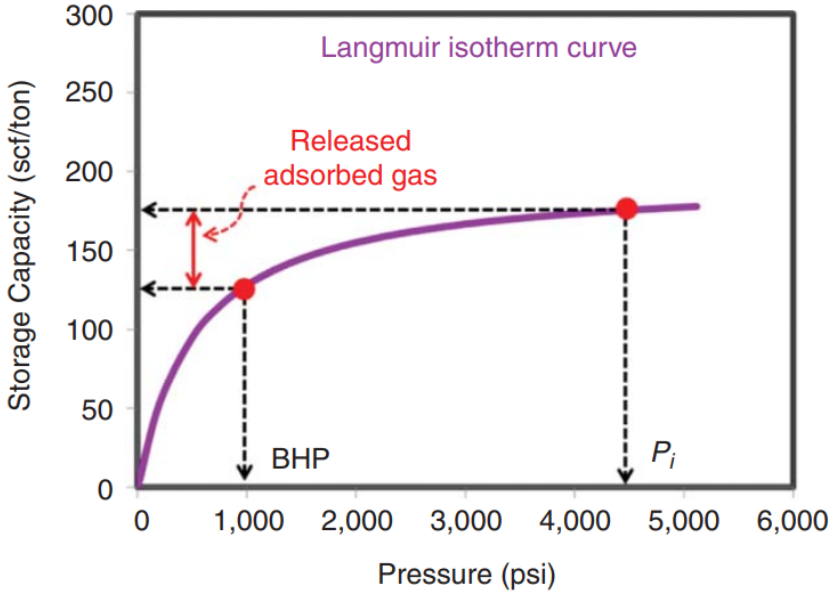


Figure 2-4. Langmuir isotherm (Yu et al., 2016)

2.2 Artificial Neural Network

The history of Artificial Neural Network (ANN) can be traced back to 1940s, when Warren McCulloch and Walter Pitts created a computational model named threshold logic. With the fast development of computational power, ANN is utilized visibly and invisibly in everyday life. Although the training times for ANNs are usually long, once finished, the use of ANNs can reduce the intensity of conventional computationally demanding methods. In automotive industry, the autopilot system relies heavily on deep learning to understand road conditions and car behaviors. In medical sciences, unsupervised ANN can be a powerful tool to classify benign and malignant tumors. In financial field, the loaner can utilize supervised ANN to study data from past costumers, thus to make better decisions. In petroleum industry, ANN is commonly used in pattern/cluster analysis, signal/image processing, control applications, prediction and correlations and optimization (Centilmen, Ertekin & Grader, 1999).

2.2.1 Artificial Neuron and Layers

Human brain is so powerful that even at 21st century scientists are not able to fully understand how it functions. Every single movement of a human body is a decision resulted after processing all the information by the brain. It is known that the smallest unit of these "decision makers" in our brains is a neuron. It is estimated that a human brain can have one hundred billion neurons. As shown in Figure 2-4, there are three main parts of a neuron: dendrites, cell body and axon.

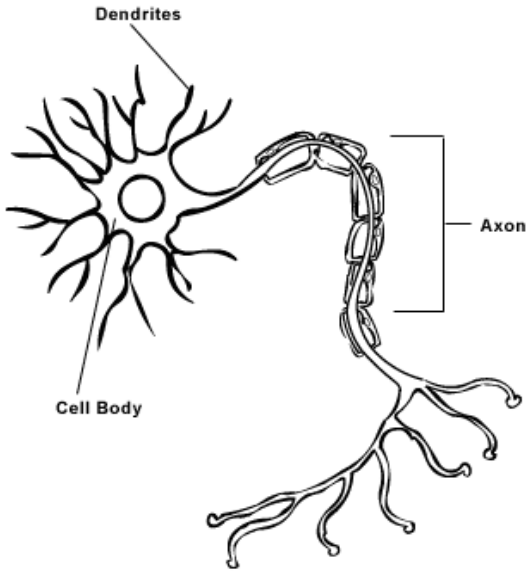


Figure 2-4. Human Brain's Neuron

Each neuron is capable of receiving stimulation from dendrites that are connected to other neurons' axons, and then process the stimulation within the cell body. The processed stimulation can then be carried by the axon to other neurons. The complex network composed by numerous neurons makes all the decisions for humans.

Artificial neural network, also known for its acronym ANN, is the intention to simulate human's brain structure using computational power. Similar to a brain's biological neural network,

a correctly structured artificial neural network is able to analyzes input values and generate predicted output. The processing is done through many artificial neurons called “nodes” placed between inputs and outputs. As illustrated by Figure 2-5, nodes are arranged in a series of layers, each of which connects to the layers on either side. Beside input layer and output layer, all other layers in between are classified as hidden layers. The connections between any two nodes are represented as “weights”. The higher the weight, the more influence one unit has on the connected unit.

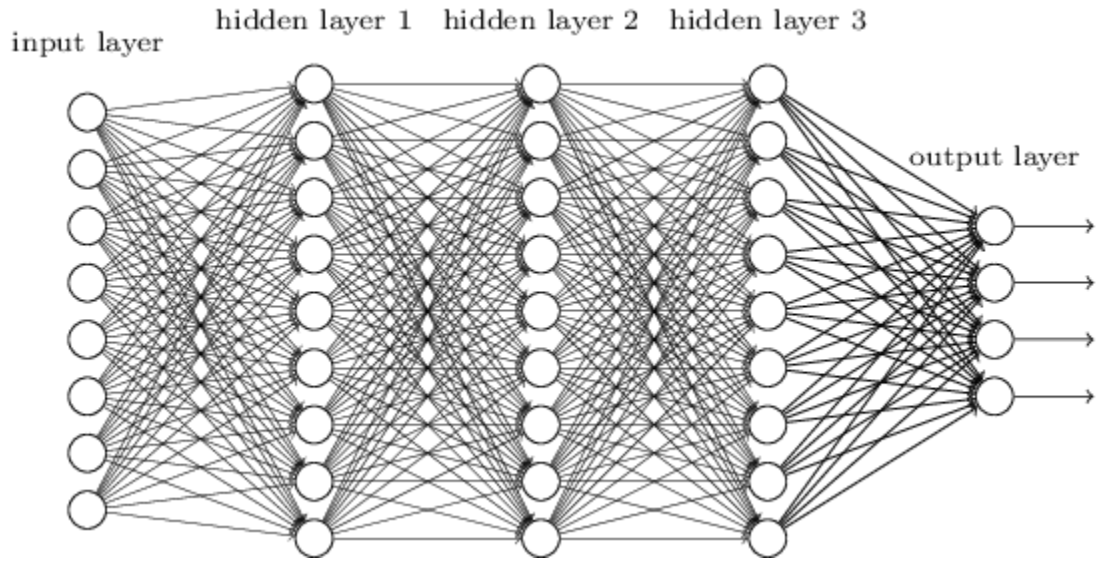


Figure 2-5. Example Feed-Forward Neural Network (Aptekar-Cassels, 2017)

2.2.2 ANN Learning Method

Similar to human’s brain, an ANN can also learn by itself. However, it is a much slower learner in comparison with human’s compatibility. ANN can be used in supervised learning and unsupervised learning. In a supervised learning, the desired output is defined before the beginning of the training. This type of training can be used to fit a complex function between two related variables. It can also be used to classify multiple type of variables. Unsupervised ANN’s target is

dynamic and changes constantly during the training period. This type of ANN is great at classifying unfamiliar data sets. Unsupervised ANN is also helpful in preprocessing and filtering the data.

The common learning process for supervised ANN is called backpropagation. It requires the information of the input layer and the output layer. This process is achieved by first randomly assigning weights on all nodes, and then comparing estimated results with actual results. If the error is significant, the weights will be updated with updating functions which use the error between two nodes as inputs. The updating process moves backward, from output layer to input layer until all the weights are updated. Then, the process will repeat until the error becomes insignificant.

For unsupervised ANN, two frequently used methods are adaptive resonance theory (ART) and self-organizing map (SOM). Adaptive resonance theory is a self-organizing neural model that tries to explain how the code is constructed. Self-organizing map utilizes a layer known as Kohonen layer that consists of processing elements (PE) which are predisposed in order. The layer of processing elements evolves during the training. The process of spatial organization is called feature mapping (Buscema, 2009).

2.2.3 Transfer Functions

Transfer functions are used between every two layers. These functions are able to manipulate input data from the previous layers. Frequently used transfer functions include hard limit function, linear function and log-sigmoid function. Common transfer functions and their input/output relations are listed in Figure 2-6.

The hard limit transfer function sets threshold to the function argument. It sets the output of the neuron to 0 if the function argument is less than 0. If the argument is greater than or equal to 0, the output is set to 1. Symmetrical hard limit transfer function is a variation of the hard limit function with its outputs being modified to -1 and 1. Hard limit transfer function is particularly

useful when it comes to classification problems. It can classify inputs into two distinct categories (Hagan, Demuth & Beale, 1996).

Name	Input/Output Relation	Icon	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

Figure 2-6. Transfer Functions (Hagan et al., 1996)

The linear transfer function is also called identity function. It simply outputs the same value as the argument. The traditional linear function does not suppress or threshold the inputs. It is commonly used between the last hidden layer and the output layer in regression problems. However, some variations such as saturating linear, symmetric saturating linear and positive linear functions incorporate bias and threshold, which can be used for specific data type.

The log-sigmoid transfer function is commonly used in between hidden layers. It squashes the output non-linearly into the range 0 to 1. Log-sigmoid transfer function is useful in

backpropagation algorithm partially because it is differentiable (Hagan et al., 1996). A common variation is the hyperbolic tangent sigmoid transfer function, which suppress the input into the range -1 to 1.

2.2.4 Training, Learning and Performance Functions

Back propagation utilizes chain rule to calculate gradients at each step. How the neural network uses the gradients is controlled by the selection of training function. Training in a neural network is equivalent to minimizing a global error function. The most used and the simplest training function is gradient descent. The error function of the gradient descent is minimized by updating weights towards the direction of the largest cost function drop. In other words, the updating of the weights follows the path with the greatest slope. Although gradient decent is easy to implement, it is considerably slower than other recent optimized training functions and not suitable for large problems.

Because this study involves training of multiplayer neural networks using large data set, conjugate gradient function is selected as the training function due to its fast convergence rate. It improves the traditional gradient decent by implementing a step size scaling mechanism. This training function effectively decreases the total training time by avoiding time consuming line search per learning iteration (Møller, 1991).

Learning function manages the update of individual weights. The learning function in gradient descent updates the weight proportionally to the derivative of the error function J:

$$w_j = w_j - \alpha \frac{\partial}{\partial w_j} J(w_0, w_1, \dots, w_n) \tag{2}$$

Where w_j is the weight at node j, and α is the dimensionless learning rate. The process is repeated until convergence.

In this study, the gradient descent with momentum weight and bias learning function is selected. It updates the weight based on neuron's input, error, learning rate and momentum constant. The equation is expressed as below:

$$dw = mc * dw_{prev} + (1 - mc)\alpha * gw \quad (3)$$

Where dw is the change of weight; gw is the gradient respect to performance, and mc is the momentum constant.

Performance function measures the goodness of a neural network. The default performance function in MATLAB Neural Network Toolbox is the mean square error function ('mse'). It is the average squared error between the actual outputs and the network outputs. In this study, mean squared error with regularization function ('msereg') is selected to be the performance function. It measures the network performance using the mean squared error, the mean squared weights and the bias values.

2.2.5 Training, Validating and Testing Set

Overfitting is one of the most common problems in neural network training. As illustrated by Figure 2-7, the well fitted function on the left captures the trend of the data while the overfitted function on the right tries to memorize specific data points. The overfitted function may have extremely low training error, but it will fail to predict accurately when new data is used.

This problem often occurs when the data set is not large enough for the selected features. Overfitting can also be caused by training for too many iterations. There are three methods to prevent overfitting: (1) generate more data; (2) reduce the number of features; (3) utilize early stopping criteria. Modifying the data set and number of features may not always be possible. Thus, incorporating early stopping criteria is highly recommended for all neural networks.

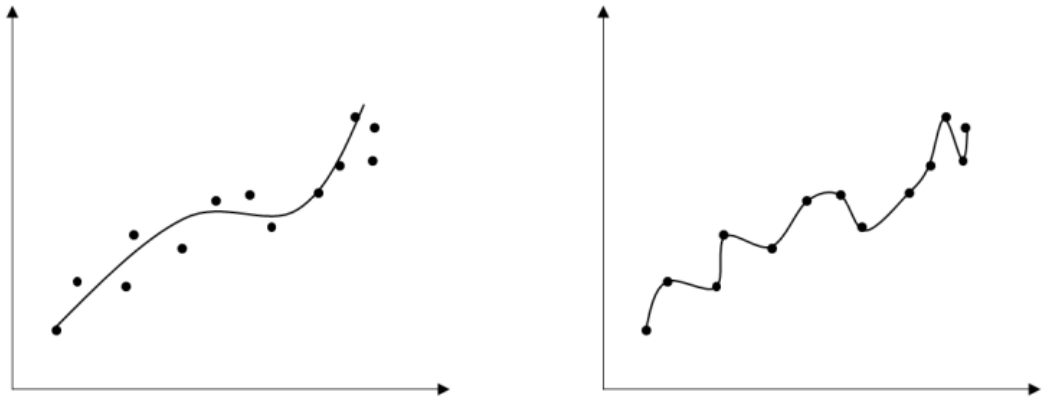


Figure 2-7. Comparison Between Well Fitted Function and Over-fitted Function (Zhang, 2017)

The first step of implementing early stopping method is to randomly divide data set into training, validating and testing set. The usual percentage ratio of each corresponding set is 70%, 15% and 15%, respectively. Training set is used solely for neural network training. Data from validation set is evaluated using the neural network at every iteration. Since the validation data is not used for training, it can monitor the ability of the neural network of predicting fresh data.

At the beginning of the neural network training, training and validating errors decrease simultaneously. The training error always continues to decrease while the validating error may start to increase after some iterations. Increasing validating error is the sign of overfitting so the network training should be terminated (Figure 2-8). Since the validating data controls the training process, it is partially biased.

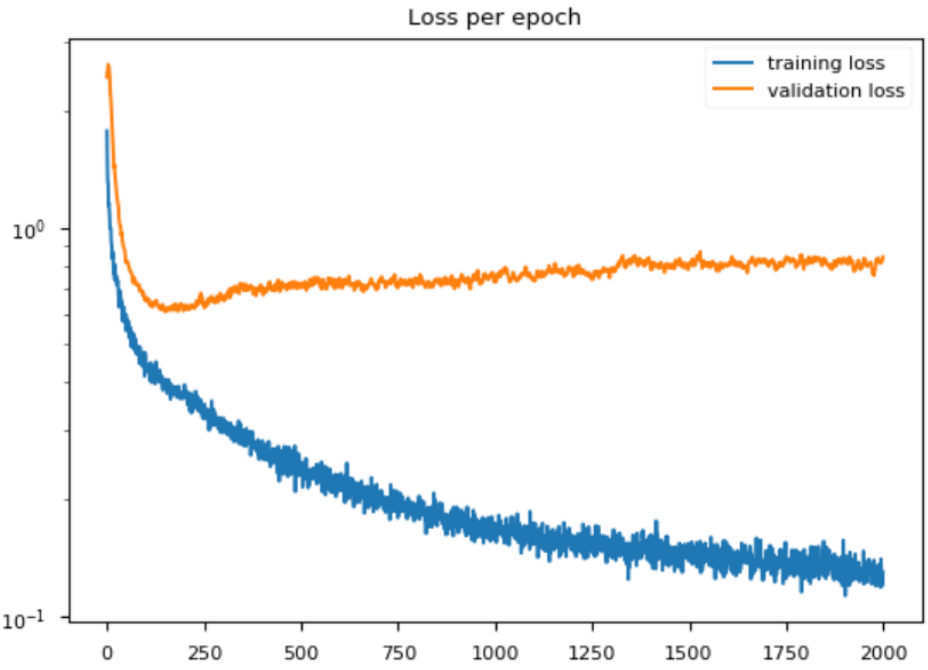


Figure 2-8. Example of Training and Validating Error with Iterations (Zhang, 2017)

On the other hand, testing set is completely unbiased. It does not participate in the training process, and it is only used to test the performance of the trained neural network. As a result, the testing set error is usually expected to be the highest among the three, and training set error tends to be the lowest.

Chapter 3

Problem Statement

Well testing is an important tool during the early development stage of a petroleum reservoir. During well testing, the pressure readings can be recorded from a pressure transducer that is installed at the bottom of the hole. Alongside the pressure data, the flowrate can also be read from the flowmeters. Analyzing the pressure and flowrate data can help the engineers to estimate critical reservoir properties such as initial reservoir pressure and rock permeability. One of the well testing techniques, pressure transient analysis (PTA) is the focus of this thesis.

The limitation of the traditional well testing approach is that it relies heavily on human's experience. In the past, petroleum engineers have produced useful analysis tools such as type curves and empirical equations. However, the accuracies of these traditional methods vary, and developing these tools is time consuming and costly.

The objective of this work is to develop an expert system that can predict reservoir characteristics within a tolerable error range based on the input PTA data. The expert system can be used in the initial stage of the field development of a shale gas reservoir after completion. The data collected during well testing period and early production period can be inputted to the expert system to estimate selected reservoir properties. Knowing properties such as porosity, permeability and fracture spacing can assist petroleum engineers to make decisions on the current well as long as future similar wells.

The expert system is composed of by seven artificial neural networks (ANN) that can be classified into two categories: forward ANNs and inverse ANNs. The forward ANN predicts the pressure transient data over 180 days period using reservoir and well completion parameters as

inputs. The inverse ANN estimates the reservoir characteristics using the PTA data and some known reservoir properties as inputs.

Chapter 4

Data Generation and Preparation

CMG-GEM, a commercial compositional reservoir simulator is used to generate data for this study. To represent a typical horizontal well in shale gas reservoir, a two-dimensional isotropic dry gas reservoir simulation model with one horizontal well at the center is developed. The visual representation is shown in Figure 4-1.

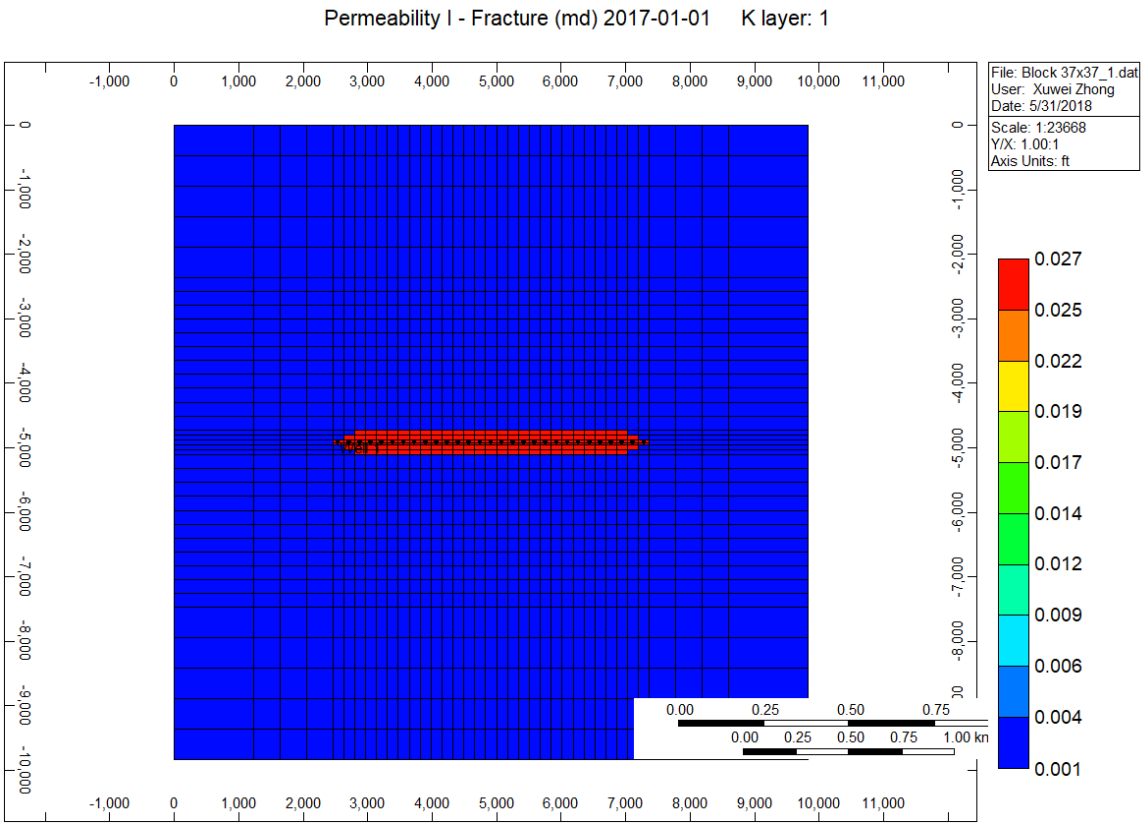


Figure 4-1. CMG-GEM Simulation Model

Since the data accuracy around the wellbore is important in this study, the simulation model requires fine grid blocks. In order to achieve this goal without sacrificing computational

cost, nonuniform grid blocks are used. The grid blocks around the well are finer than the outer regions.

In order to capture the hydraulic fractures, the Stimulated Reservoir Volume (SRV) approach was implemented. The stimulated area is marked in red as shown in Figure 4-1. The elliptical shape represents the fracture zone of a typical tight gas well. At the SRV blocks, the fracture permeability increases and the fracture spacing decreases.

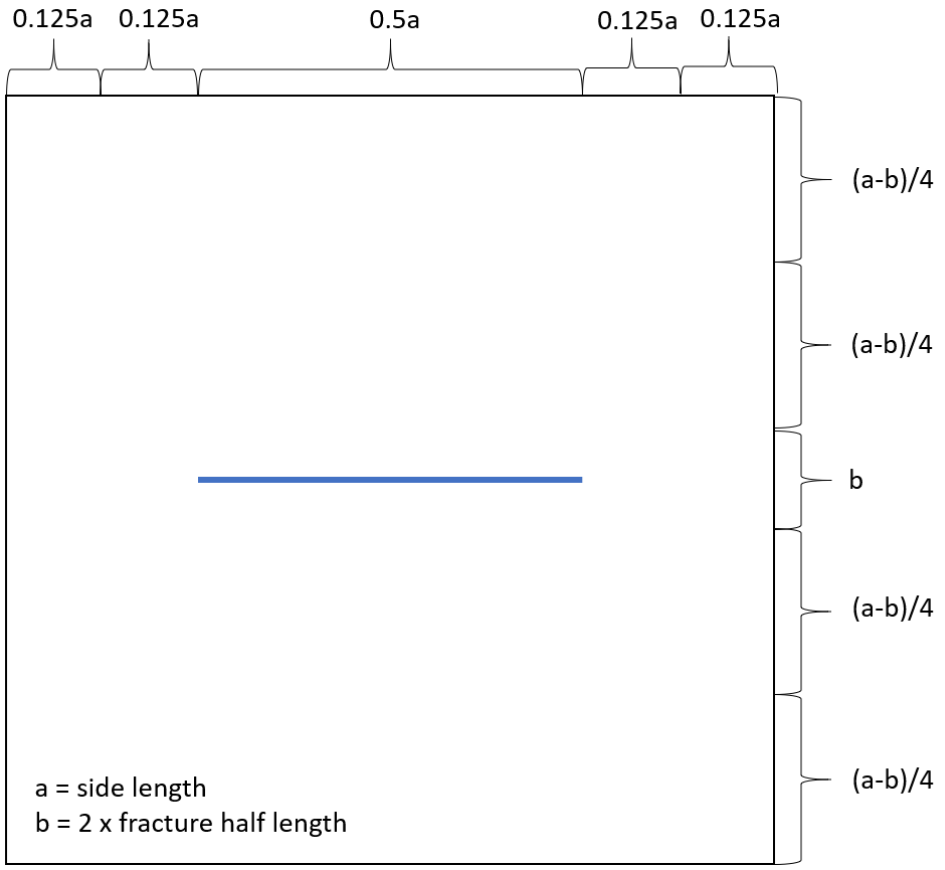


Figure 4-2. Grid Block Size Distribution

The determination of grid block sizes is directly related to the length of the horizontal well as shown in Figure 4-3. In order to prevent pressure disturbance reaching the boundary due to the well production, the side length of the model is always twice the length of the well. The remaining

areas are divided as illustrated by Figure 4-2. More grid blocks are assigned to the locations close to the wellbore.

4.1 Grid Blocks Sensitivity Analysis

The density of the grid blocks greatly impacts the simulation results. Theoretically, the model with infinite number of grid blocks produces the most accurate results, but it is impossible for computers to finish the task. Thus, the tradeoff between computational cost and the marginal increment in accuracy was investigated. The goal was to select the number of grid blocks that could produce accurate results, while having a reasonable run time. All input parameters were kept constant and the values are shown in Table 4-1.

Table 4-1. Input Parameters for Grid Blocks Sensitivity Analysis

Reservoir Parameter	Value
Grid Top Depth	6500 ft
Thickness	150 ft
Porosity	0.1
Porosity - fracture	0.01
Permeability-matrix	0.0005 md
Permeability-fracture	0.025 md
Fracture Spacing	2 ft
Pressure	3940 psi
Rock Compressibility	1e-5 1/psi
Reservoir Temperature	142 F
Gas Density	0.0416 lb/ft3
Langmuir Volume of CH4	70 scf/ton
Langmuir Pressure of CH4	750 psi
Flow Rate	1 mmscf/day

A grid block sensitivity analysis was conducted by building multiple simulation models using the same configuration as shown in Figure 4-2. Table 4-2 summarizes the simulation time corresponding to different number of grid blocks. Figure 4-3 shows the bottom hole pressure decline curves for the first 180 days using different models. As illustrated by the figure, as the number of grid blocks increases, the pressure decline curve moves upwards and the marginal improvement in accuracy decreases. The 37x37 grid block system was selected after comparing its decline curve against the decline curve of the 51x51 grid block system. Two curves are almost identical with minimum differences, while the 37x37 system shows significantly lower simulation time.

Table 4-2. Grid Block Number and Corresponding Time of Simulation

Number of Grid Block	Time of Simulation(s)	Time of Simulation for 1000 cases (hr)
21x21	10	2.78
25x25	12	3.33
27x27	13	3.61
33x33	18	5.00
37x37	21	5.83
51x51	30	8.33

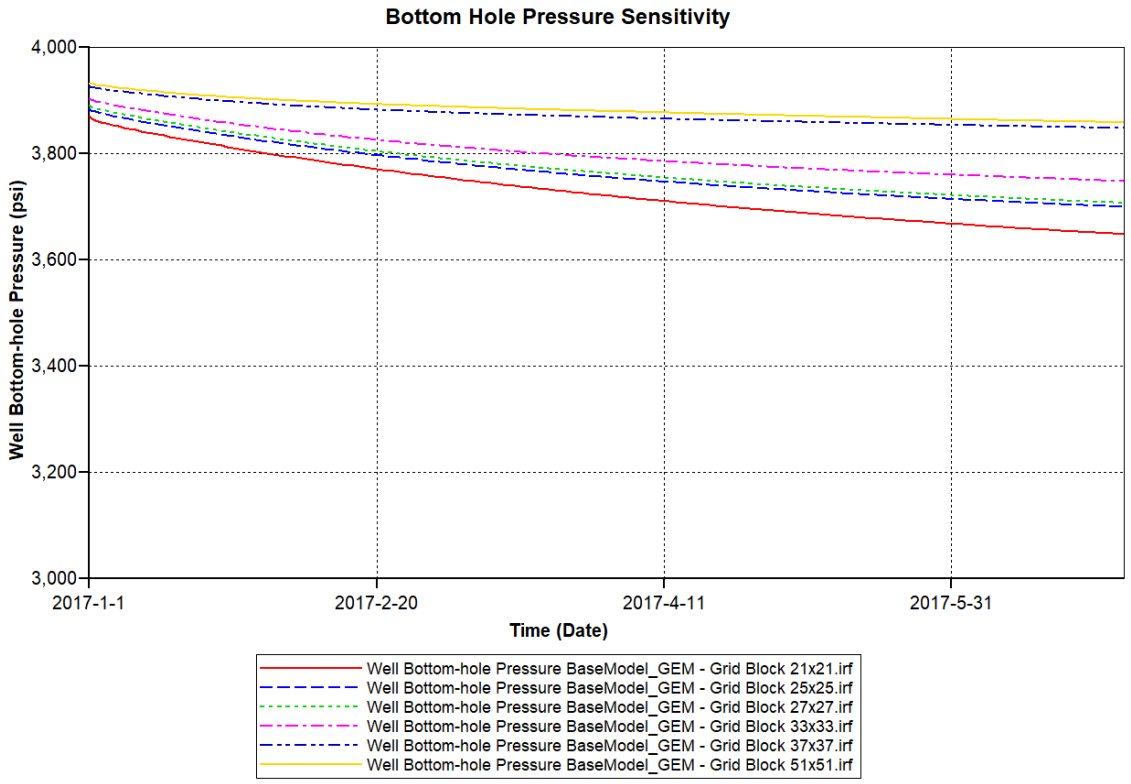


Figure 4-3. Bottom Hole Pressure Curve for Different Grid Block System

4.2 Range of Parameters

Training an artificial neural network requires a large data set. The more cases the neural network learns, the better it predicts. Thus, thousands of different simulation models need to be built. 15 input variables including reservoir characteristics and well completion parameters were selected, and their ranges are shown in Table 4-3. The values of the parameters were selected randomly within the assigned ranges when building simulation models. With large enough data set, this approach can ensure that most combinations of the input parameters are trained by the expert system. All other required parameters in the model were set to be constants. The simulation period for each run was set to be 180 days.

Table 4-3. Simulation Model Input Parameter Range

Simulation Model Parameter	Value Range
Thickness	50 ~ 200 ft
Porosity - matrix	0.06 ~ 0.11
Porosity - fracture	0.006 ~ 0.015
Porosity - SRV	0.06 ~ 0.15
Permeability - matrix	0.00001 ~ 0.00009 md
Permeability - fracture	0.0003 ~ 0.009 md
Permeability - SRV	0.003 ~ 0.09 md
Fracture Spacing	1.5 ~ 4 ft
Fracture Spacing - SRV	0.15 ~ 0.4 ft
Fracture Half-Length	50 ~ 200 ft
Langmuir Volume of CH4	50 ~ 150 scf/ton
Langmuir Pressure of CH4	500 ~ 1000 psi
Well Length	3000 ~ 5000 ft
Reservoir Pressure	2000 ~ 4000 psi
Flow Rate	1 ~ 10 mmscf/day

4.3 Data Preparation

Once the parameter range was decided, the data set could be generated with the help of MATLAB scripts. The DAT file used in CMG-GEM software includes all information of the simulation model. In order to generate multiple DAT files, the first step was to generate a template. The template consisted all the required constant values. All 15 input parameters and variables that could change because of the input parameters were substituted by different keywords. Simple MATLAB scripts were written to first randomize the input parameters, then substitute the keywords

with generated numbers. Upon completion, a new complete DAT file was saved. Repeating this process with a for-loop could efficiently generate large number of cases.

In order to produce results, all generated DAT files need to be run using CMG-GEM. The software is able to queue jobs to save users' time. Thus, all DAT files were added in the queue and the corresponding models were simulated in order. In CMG Suite, the produced results are stored in IRF files. For the purpose of this study, the simulation results need to be stored in numerical tables. The CMG-Results Report software was used to access IRF files by taking RWD files as inputs. The RWD file indicates the IRF file that is going to be imported, and it informs the software the variables that need to be stored. Similar to DAT files generation, a template for RWD file was created. Bottom hole pressure data was produced by using MATLAB and CMG-Results Report software. All relevant scripts and files can be found in the appendix section of the thesis.

Pressure transient curve was approximated using seven data points. Although having more data points could smoothen the decline curve, excessive points could increase the complexity of the neural networks and deteriorate the overall accuracy. Seven points were selected based on the time ($t_{BHP=14.7\ psia}$) for the bottom-hole pressure of each case to reach lower limit. If the pressure did not decrease fast enough to reach atmospheric pressure before the end of simulation, the time would be set to equal to 180 days. Seven pressure points were recorded as the following expressions: $P_{wf@day\ 1}$, $P_{wf@1/6t}$, $P_{wf@2/6t}$, $P_{wf@3/6t}$, $P_{wf@4/6t}$, $P_{wf@5/6t}$ and $P_{wf@6/6t}$. Fraction of the time was rounded to the nearest whole number. If the bottom-hole pressure of the case did not reach 14.7 psia in 180 days, the pressure data would be read at day 1, day 30, day 60, day 90, day 120, day 150 and day 180. Otherwise, the pressure data would be read at first day and six other calculated days.

Chapter 5

Development of Artificial Neural Network

In this study, the assembled expert system consists of seven artificial neural networks (ANN). As shown in Figure 5-1, the forward ANN includes ANN0, ANN1 and ANN2. The inverse solution is composed by ANN3a, ANN3b, ANN4a and ANN4b. All the ANNs were built using MATLAB Neural Network Toolbox.

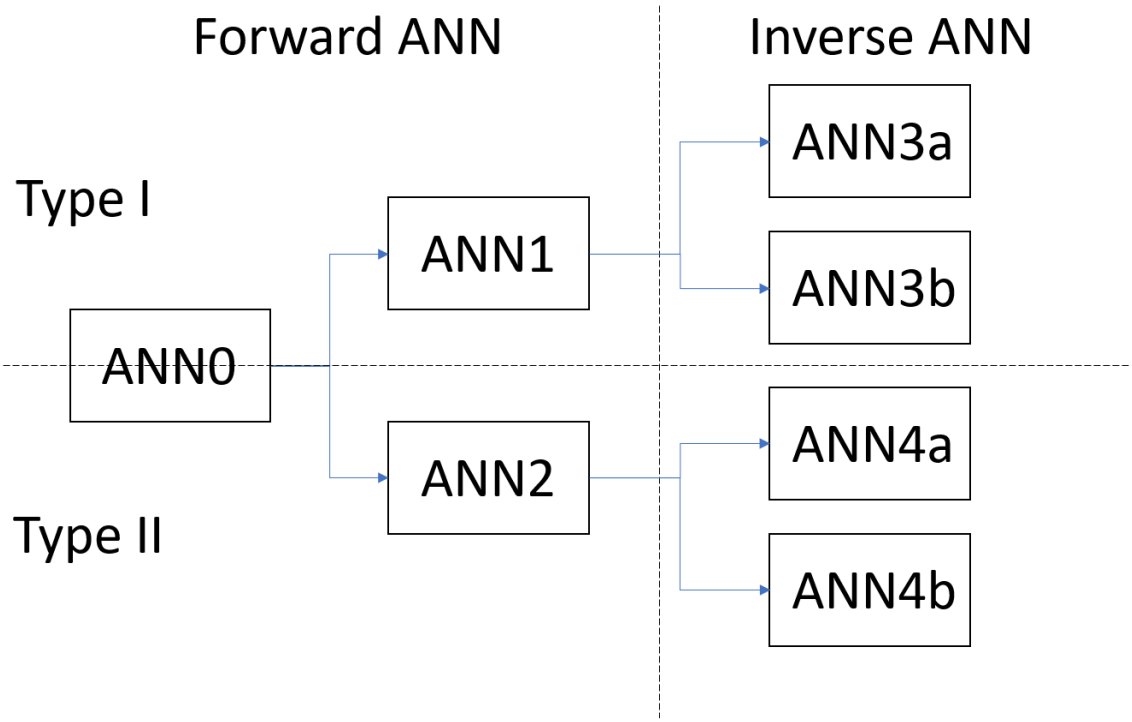


Figure 5-1. Overview of ANN Expert System

During the data generation stage, two types of pressure transient curves were observed. The more common type had smooth decline curve with decreasing magnitude of slope as time increased (Figure 5-2). The other type had rapid decline rate, and the bottom hole pressure reached the lower limit (atmospheric pressure) before the simulation ended (Figure 5-3). It had a

discontinuity point which could hinder the performance of the expert system. However, since the pressure readings after the discontinuity point were constant, only the declining portion need to be learned by the expert system.

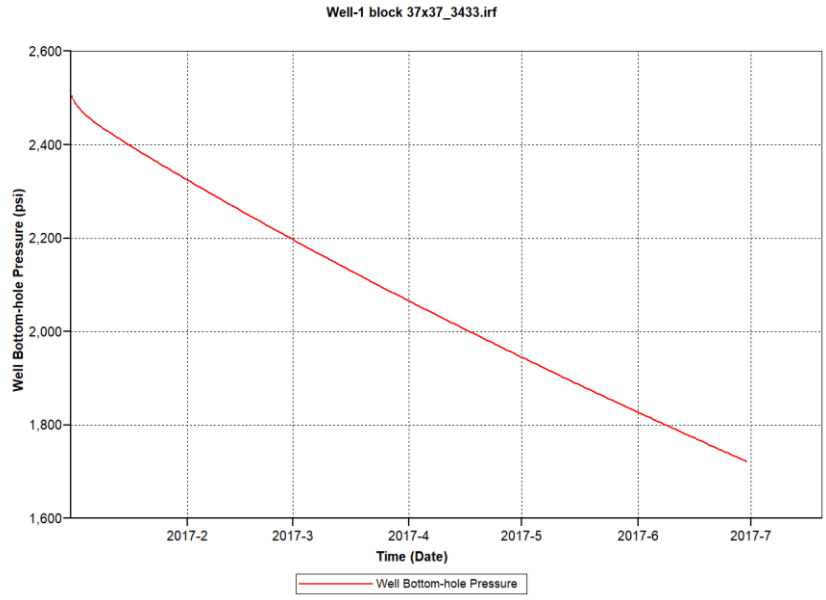


Figure 5-2. Example of Type I Pressure Decline Curve

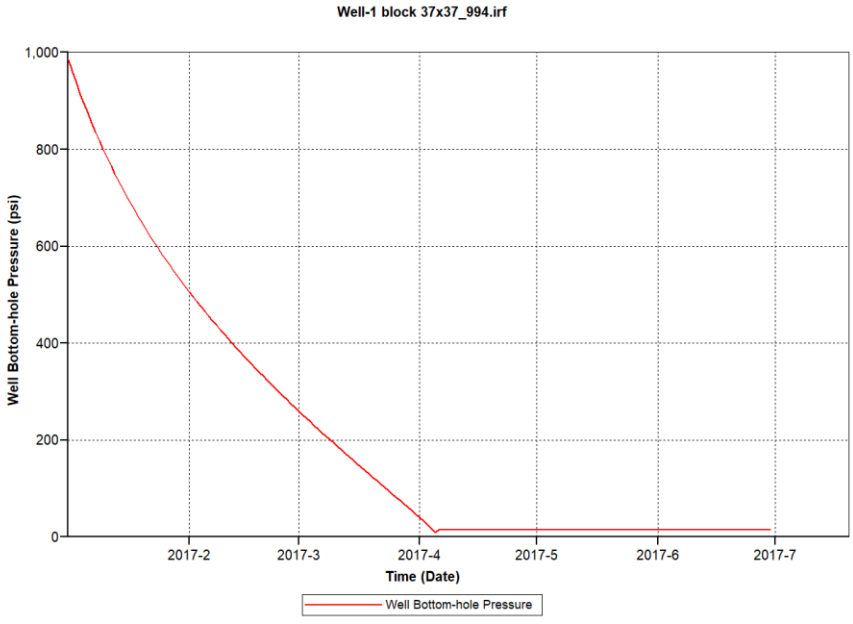


Figure 5-3. Exam of Type II Pressure Decline Curve

Data were classified into two types as illustrated by Figure 5-1. The corresponding ANNs were trained using appropriate data sets. Although feedforward neural networks with multilayer perceptron has been widely used, recent results have showed that the fully connected cascade feedforward neural network is more powerful (Qiao et al., 2015). This architecture allows connection between every two layers and improves the accuracy. Thus, all ANNs in the expert system were built as cascade feedforward neural networks.

ANN0 is a classification neural network that identifies two different types. ANN1 and ANN2 are neural networks that produce pressure transient data based on reservoir characteristics and well completion parameters input. ANN3a and ANN4a are tools that estimate reservoir characteristics. ANN3b and ANN4b are validation tools that can are used in closure test which checks the internal consistency of the expert system.

5.1 Data Screening

Initially, 7000 cases were generated using different combinations of randomized variables as indicated in Table 4-3. However, there were less than 1000 type II cases, and the majority belongs to type I. In order to minimize the effect of overfitting, especially in type II ANNs, another 7000 cases were created. As a result, the raw dataset had 13306 type I cases and 2006 type II cases.

However, the raw dataset included some undesirable and unphysical cases which need to be removed. Firstly, some cases had combinations of low initial pressure, high flow rate and low thickness. The flow rate was overly high for the reservoirs with poor storage characteristics. Thus, not only did such cases fall into type 2, but also many of them reached the minimal pressure on the first day of production. The collected pressure data from these cases had multiple points with constant minimal values. There were also few extreme cases which landed 14.7 psia immediately after being put on production. The recorded data were seven constant points that need to be deleted.

Secondly, negative pressure values were saved in the raw data. This unphysical behavior was due to the selection of time step in the numerical model. Since the model uses constant flow rate as the production constraint, pressure drops are expected at every time step. For some cases, the pressure drops may exceed what the model can provide and result in new bottom-hole pressures which are lower than 14.7 psia. The CMG-GEM software auto-corrects this unphysical behavior by changing the production constraint to constant bottom-hole pressure in the next time step. Thus, the flow rate decreases after the change of constraint because the available pressure drop is limited by the atmospheric pressure.

To combat the occurrence of unsatisfactory pressure transient data, only cases with $t_{BHP=14.7\ psia}$ greater and equal to 6 days were saved. This technique ensured the declining nature of the output data. In addition, all pressure data smaller than the atmospheric pressure were manually adjusted to be equal to 14.7 psia. After screening, there were 11300 type 1 cases and 1982 type 2 cases available for the expert system.

5.2 Forward Neural Network

The first step of this research was to build a system that can take known reservoir properties and well completion parameters information, and then generate pressure transient curves. The idea is illustrated by the equation in Figure 5-4. The plus sign implies the combination of reservoir property data and well completion parameters. All the information can then be processed by the neural networks. The produced results from the expert system "equals" to the pressure transient data. Since the pressure transient data can be generated by reservoir simulator directly, the forward ANN may not have practical use. However, it is the first step towards the development of inverse solution which is the objective of this study.

The forward neural network is consisted of ANN0, ANN1 and ANN2. ANN0 is the classification neural network that can categorize data into type I and type II. ANN1 and ANN2 are pressure transient data prediction neural networks for two data types accordingly.

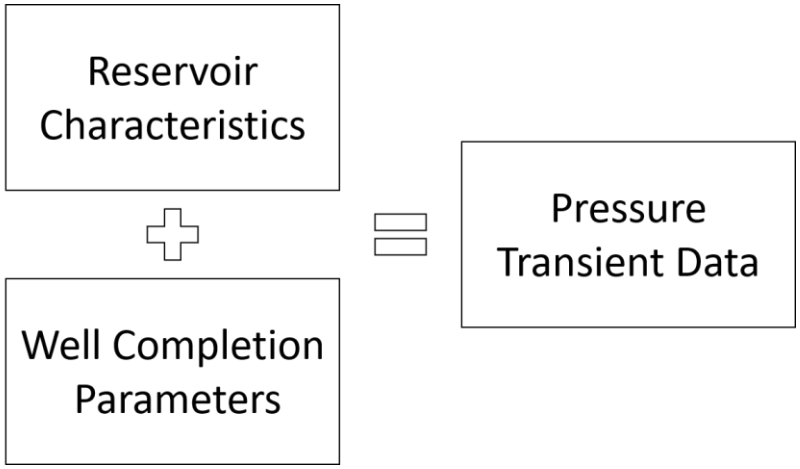


Figure 5-4. Forward Solution Diagram

5.2.1 Development of ANN0

ANN0 is a neural network that intakes reservoir and well completion parameters data and predict time required for each case to reach 14.7 psia. It also works as a classification neural network because the of the output. Cases with ANN0's results lower than 180 days are classified as type I. Cases with output data equal to 180 days fall into type II.

The first step of developing neural networks was to understand the complexity of the problem. In our study, ANN0 was designed to have 15 input parameters as shown in Table 4-3. The only output was the time required to reach minimal pressure. Although ANN0 only included one output neuron, having type I and type II cases in one neural network increased its complexity.

The structure of ANN was determined based on trial and error and rule of thumb. Since the number of hidden layers had largest impact on neural network training time, it was tested first.

After deciding on the number of hidden layers, different number of neurons in each layer and different transfer functions between layers were examined. The determination of the number of neurons used rule of thumb along with random guess. One rule of thumb is that the number of hidden neurons should be between the size of the input layer and the size of the output layer (Heaton, 2008). The trial and error process terminated when no improvement in accuracy was discovered. The training process using MATLAB Neural Network Toolbox is illustrated in Figure 5-4. Early stopping was implemented, and the program automatically stored the data at the epoch with the lowest validation error. The training stopped when the neural network reached goal performance or the maximum epoch set by the user.

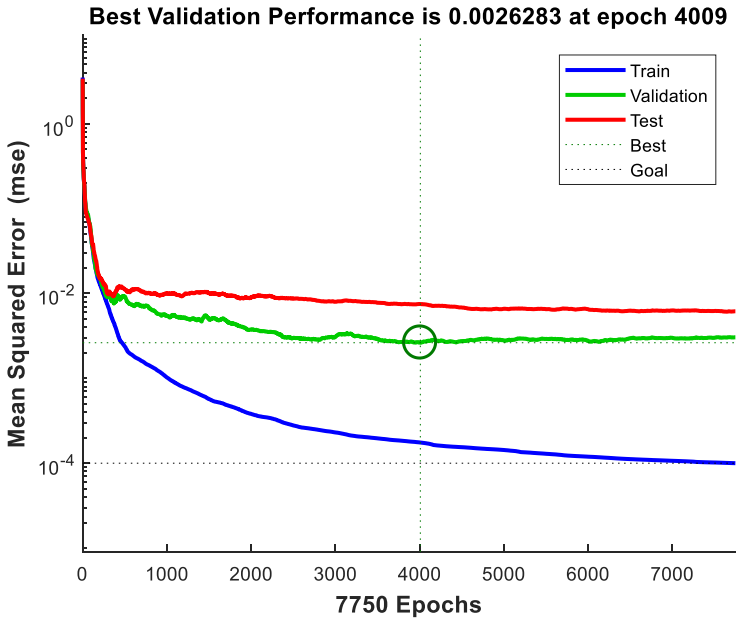


Figure 5-4. Example of Neural Network Training in Progress

Initially, simple ANN with one hidden layer was trained and tested. The training was rapid due to the simplicity of the ANN while sacrificing minor accuracy. Next, more hidden layers were tested. It was discovered that ANN0 had the best prediction when it had three hidden layers. Then, the number of neurons and transfer functions were examined using random guess and rule of thumb.

The optimal structure was found to be three hidden layers with seven neurons in each layer. Hyperbolic tangent sigmoid transfer function was used between the layers and linear transfer function was used between the output layer and the output node. The architecture of ANN0 is shown in Figure 5-5.

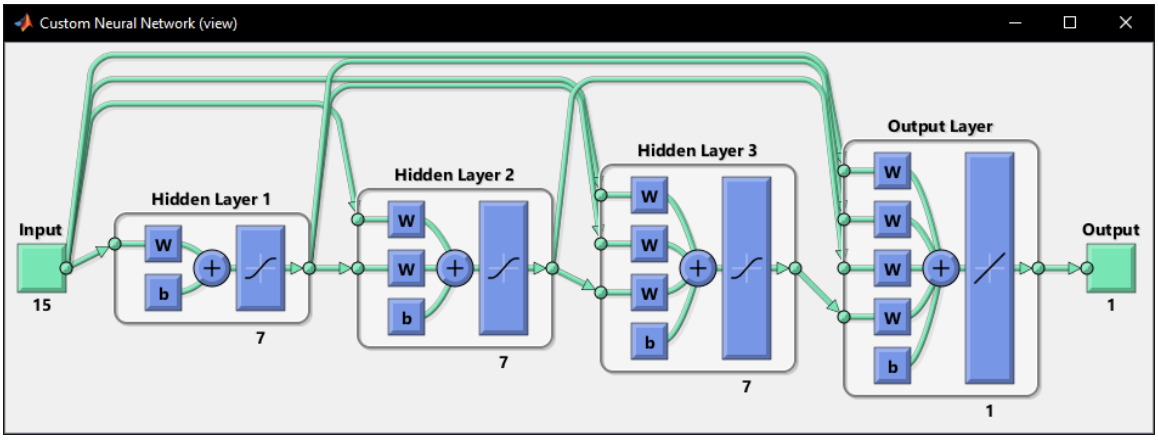


Figure 5-5. ANN0 Architecture Diagram

After the architecture of ANN0 was decided, the input and output data were suppressed to values between -1 and 1. This range worked the best with hyperbolic tangent sigmoid transfer function since they shared the same upper and lower limits. Data suppression was able to bring all data into the same magnitude. The data scaling was especially important in this study because the data set contains parameters at wide ranges. Having matrix permeability at values as low as $10^{-5}\ md$ and flow rate at values as high as $10^{6}\ \frac{scf}{day}$, the unscaled data set could add confusion to the neural network.

Neural network weights were randomly assigned when the training started. In addition, the data set was divided randomly into training, validation and testing sets. As a result, the resulting neural network became slightly different every time it is trained. In order to obtain the most optimized results, the neural network was trained multiple times, and the one with the lowest error was stored.

5.2.2 Development of ANN1 and ANN2

ANN1 and ANN2 are neural networks that intake reservoir and well properties and predict pressure transient data. The neural networks are used for type I and type II data accordingly.

Both neural networks were designed to include the same 15 parameters as indicated in Table 4-3. The input parameters were reservoir thickness, matrix porosity, fracture porosity, SRV fracture porosity, matrix permeability, fracture permeability, SRV fracture permeability, fracture spacing, SRV fracture spacing, fracture half-length, Langmuir volume of CH4, Langmuir pressure of CH4, well length, reservoir pressure and flow rate. Since the type II pressure decline curve reached the atmospheric pressure before the simulation ends, ANN2 included an additional parameter: time to reach minimum pressure. Seven output nodes were seven pressure points: $P_{wf@day\,1}$, $P_{wf@1/6t}$, $P_{wf@2/6t}$, $P_{wf@3/6t}$, $P_{wf@4/6t}$, $P_{wf@5/6t}$ and $P_{wf@6/6t}$ as explained in Chapter 4.

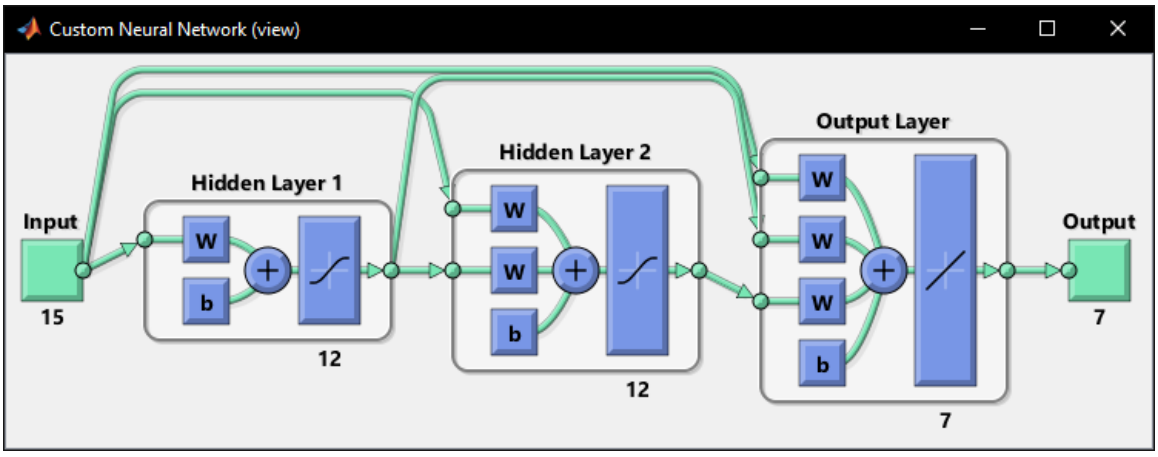


Figure 5-6. ANN1 Architecture Diagram

The architectures of ANN1 and ANN2 were determined based on trial and error and rule of thumb. The selection of structures followed the same process as described in the previous section. The complexity of the neural networks was increased until the improvement of performance was trivial. It was found out that two hidden layers was sufficient for forward solution

neural networks likely due to the simple declining trend among output points. The architectures of ANN1 and ANN2 are illustrated by Figure 5-6 and Figure 5-7. ANN1 had two hidden layers with twelve neurons in each layer. ANN2 had similar structure with two hidden layers and ten neurons in each layer.

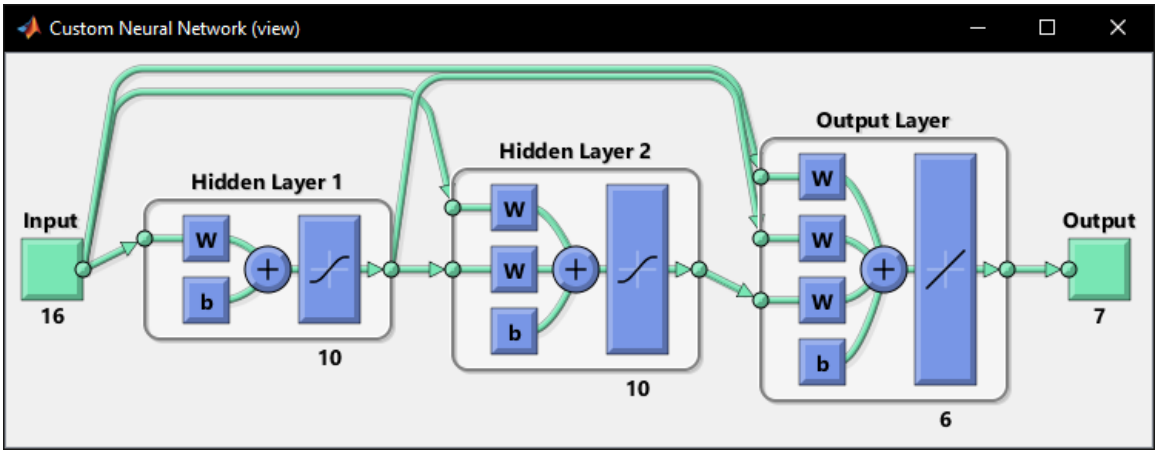


Figure 5-7. ANN2 Architecture Diagram

Similar to ANN0, input data were suppressed to the same magnitude before the training. Hyperbolic tangent sigmoid transfer and linear transfer function were used between the layers. The best performed neural network was stored after multiple training sessions.

5.3 Inverse Neural Network

The second step of this research was to build a system of neural networks that can estimate reservoir characteristics based on well production profile. If trained successfully, the expert system can be a good alternative method for history matching problems which are common in petroleum engineering.

The inverse neural network is consisted of ANN3a, ANN3b, ANN4a and ANN4b. ANN3a and ANN4a are reservoir characteristics prediction tools for type I and type II data accordingly.

ANN3b and ANN4b are validation tools that generate well completion parameters. The reservoir characteristics prediction tool is the main objective of this study. The user can estimate reservoir properties based on existing well information and well testing results. The well completion parameter validation tool is less practical and mainly serves as one of the components in closure test of the expert system.

5.3.1 Development of ANN3a and ANN4a

ANN3a and ANN4a are neural networks that use pressure transient data and well completion parameters as inputs to predict reservoir characteristics. The idea is illustrated by the equation in Figure 5-8.

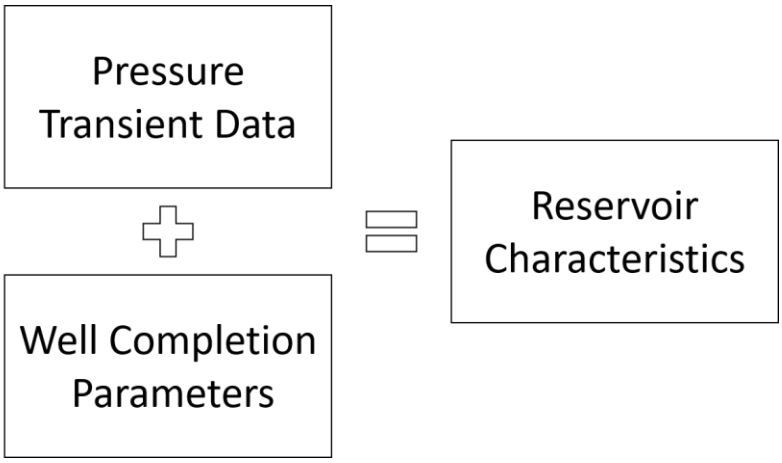


Figure 5-8. Inverse Solution Diagram for Reservoir Characteristics Prediction Tool

Input and output parameters were decided first. Although the expert system was designed to predict reservoir characteristics, it was unrealistic to build neural networks that can estimate all the reservoir properties using only seven pressure points and few well completion parameters. Thus, only the most important reservoir characteristics were selected as outputs, and their ranges can be

looked up from Table 4-3. The outputs were matrix porosity, fracture porosity, SRV fracture porosity, matrix permeability, fracture permeability, SRV fracture permeability, fracture spacing and SRV fracture spacing. These characteristics were selected because they had large influence on production profile, and they were more challenging to be assessed using traditional well logging methods. Properties such as reservoir thickness, reservoir pressure, Langmuir volume and pressure of CH4 were assumed known. Moreover, the seven pressure points and well completion parameters were added to input parameters. The well completion parameters were well length, flow rate and fracture half-length. ANN4a which worked with Type II data included the extra input parameter: time to reach minimal pressure.

Similar to the forward neural network development, the architectures of ANN3a and ANN4a were determined using trial and error process. It was found that both networks had the best performance with two hidden layers. The number of neurons and the type of transfer functions had minimal to no impact on the accuracy of the neural networks. In fact, the randomness of data division into training, validation and testing set had larger impact than the structure of hidden layers, especially for ANN4a. Since ANN3a was trained using large data set, the accuracy was more stable and less influenced by the random data division than ANN4a. The input and output data were suppressed to the same magnitude, and the neural network with the lowest error was stored after multiple training sessions.

Figure 5-9 and Figure 5-10 show the architectures of ANN3a and ANN4a. The number of neurons in ANN4a's hidden layers were smaller than ANN3a due to difference in the size of data set. Overfitting could happen if the data set was not large enough for the complexity of the neural network.

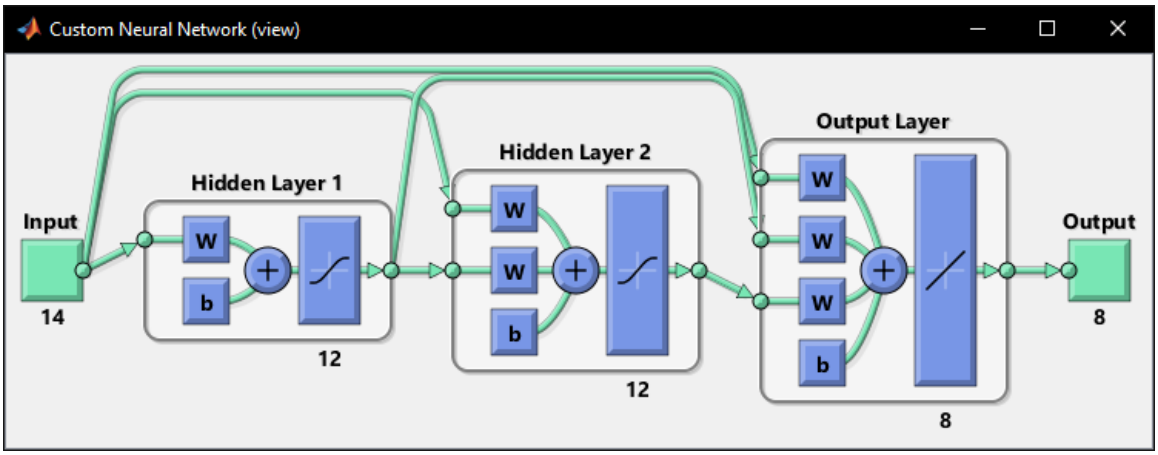


Figure 5-9. ANN3a Architecture Diagram

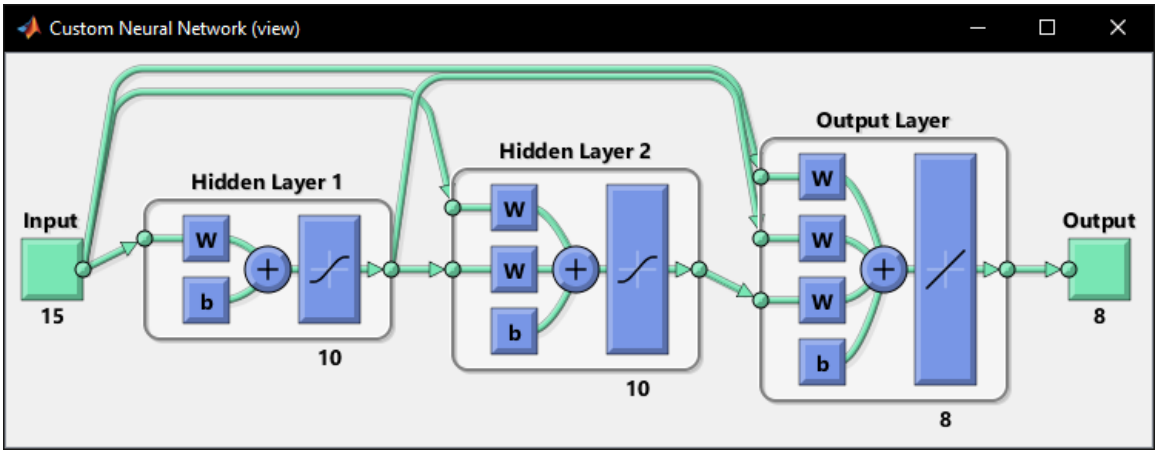


Figure 5-10. ANN4a Architecture Diagram

5.3.2 Development of ANN3b and ANN4b

ANN3b and ANN4b are validation neural networks that generate well completion parameters. The validation neural networks close the loop of the three major components: pressure transient data, reservoir characteristics and well completion parameters. They can produce well completion parameters such as well length, flow rate and fracture half-length based on reservoir characteristics and well production profile. Figure 5-11 illustrate the idea of this inverse solution.

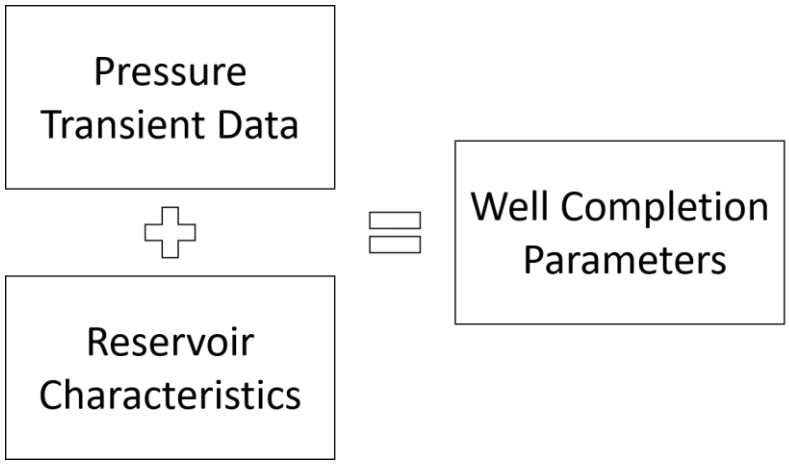


Figure 5-11. Inverse Solution Diagram for Well Completion Parameters Validation Tool

Since the well completion parameters are known before the well test in real fields, it is unpractical to build a neural network that predicts these parameters. Thus, the main purpose of the well completion parameter prediction tool is to validate the goodness of the expert system. The validation is conducted through closure test discussed in section 6.3.

Reservoir characteristics and pressure transient data were assumed to be known and were used as inputs. The inputs included seven pressure points, matrix porosity, fracture porosity, SRV fracture porosity, matrix permeability, fracture permeability, SRV fracture permeability, fracture spacing, SRV fracture spacing, reservoir thickness, reservoir pressure, Langmuir volume of CH_4, Langmuir pressure of CH_4. ANN4b also included time to reach minimal pressure as one of the inputs. The outputs were well length, flow rate and fracture half-length.

Similar to other ANN development, trial and error method was used to determine the architectures of ANN3b and ANN4b. It was found that ANN3b had the best accuracy with two hidden layers and eight neurons in each layer. The error was stable with minor variation due to the large data set. In contrast, ANN4b's error changed greatly from the random data division. It was discovered that at the number of neurons smaller than ten in each layer, the neural network could

be trained longer without encountering overfitting early. After deciding the architectures (Figure 5-12 and Figure 5-13), the neural networks were trained multiple times in order to achieve the best accuracy.

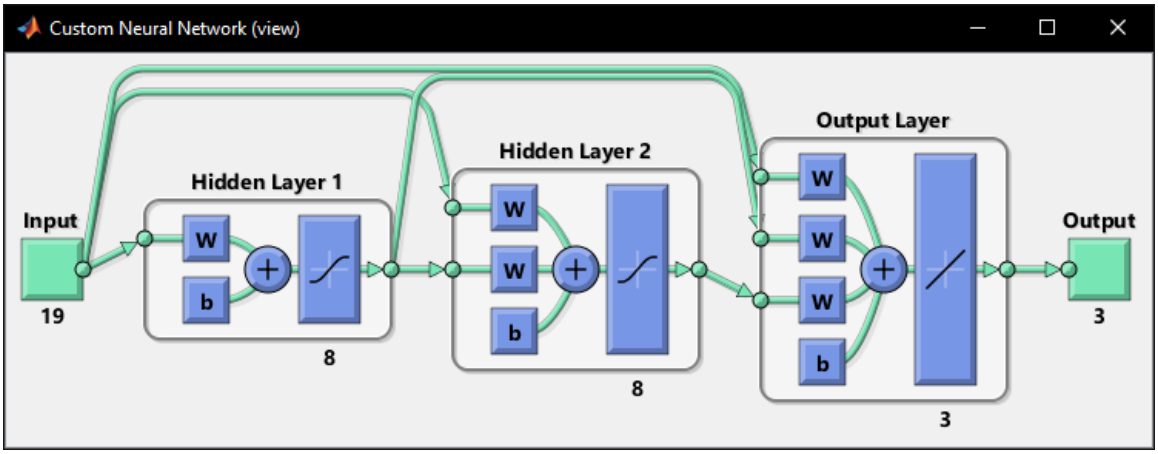


Figure 5-12. ANN3b Architecture Diagram

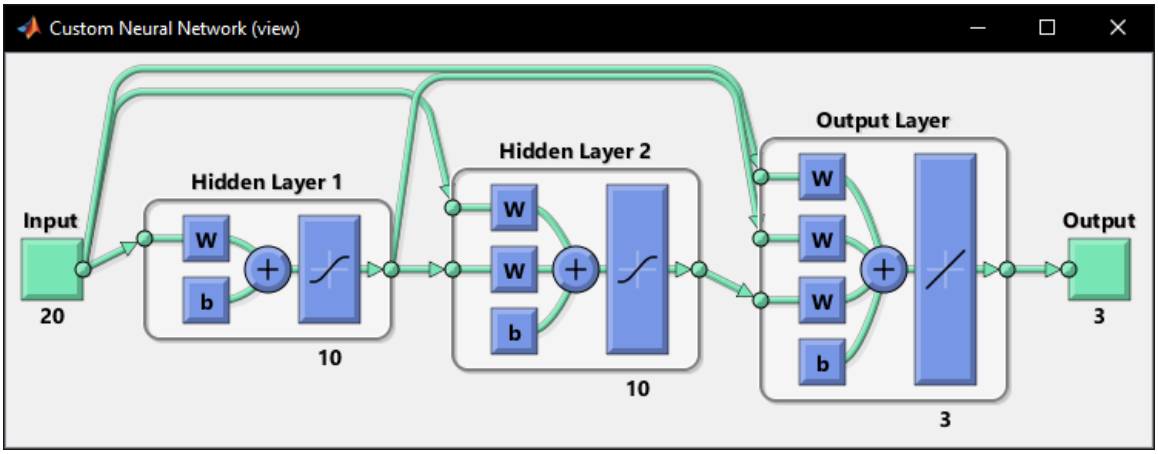


Figure 5-13. ANN4b Architecture Diagram

Chapter 6

Results and Discussions

In this chapter, the results of forward solution, inverse solution and cosure test will be discussed and analyzed. The error rate is calculated using the following equation:

$$Error = \frac{|estimated\ value - actual\ value|}{actual\ value} \times 100\% \quad (4)$$

Two types of data are used in training of seven ANNs included in the expert system. The results from ANNs using type I data are nearly identical to the results generated from ANNs using type II data. Due to the more complicated nature and smaller size of type II data, the errors of type II ANNs are almost always greater than the errors from type I ANNs. The difference in percentage error between two ANN types is usually smaller than 5%. Thus, to avoid repeating, only type II ANNs results will be discussed in detail in this chapter. Results, tables and figures for type I data can be found in Appendix D.

6.1 Forward ANN Results

The forward ANN includes ANN0, ANN1 and ANN2. Results on only type II ANNs (ANN0 and ANN2) will be discussed here. Average percentage error for each ANN is presented along with error distribution of all output parameters. For ANN2, the best case and worst case of pressure transient data predictions are selected and graphed. Overall, forward ANN can predict output parameters accurately with less than 3% error rate. The trend and values of bottom-hole pressure data are correctly captured.

6.1.1 ANN0 Results

Table 6-1 displays the average numerical and classification error of ANN0. As excepted, the training error is smaller validation and testing errors. The testing error is as little as 1.58%, which shows that the ANN0 can predict a fresh data set precisely. In addition, the classification error of ANN0 is even smaller at 0.7% for the testing set. It will be a rare occasion for new data to be categorized into wrong data type.

Figure 6-1 illustrates the error distribution of ANN0 with histogram and pie chart. The testing set has 1992 cases in total. 92% of the predictions have smaller than 5% error rate, and only 2% of the cases have greater than 20% error rate. It is discovered that the cases with outputs close to the boundary tend to have high error rates. Cases when pressure decreases immediately to the minimum and the cases that reach the atmospheric pressure at nearly 180 days have error rates as large as 50%.

Table 6-1. ANN0 Average Error

	Average Numerical Error	Average Classification Error
Training Set	1.05%	0.09%
Validation Set	1.67%	1.10%
Testing Set	1.58%	0.70%

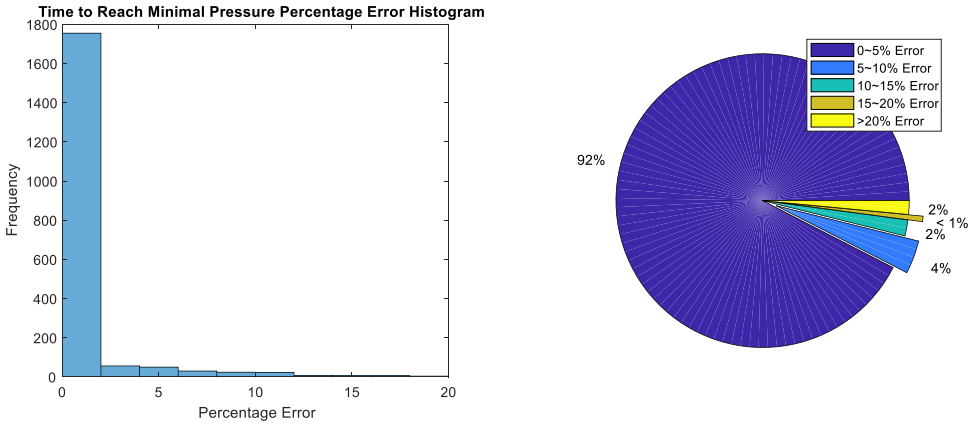


Figure 6-1. ANN0 Time to Reach Minimal Pressure Error Distribution

6.1.2 ANN1 and ANN2 Results

ANN1 and ANN2 have similar performance. ANN1 has slightly more accurate predictions with 1.3% testing set error, and the detailed results can be found in Appendix D. Only ANN2 results are discussed in this section.

Table 6-2 shows the average error of ANN2.The performance is good and error rates for training, validation and testing sets are all below 3%. Figure 6-2 shows the error distribution of the estimated bottom-hole pressure.

The testing set has 297 cases in total. Note that seven output pressure points are treated as one parameter, so the figure shows the total error distribution of 2079 pressure data points. Around 1600 data points have error rates smaller than 2%, and 91% of the testing data have smaller than 5% error rates. Only two cases out of 2079 cases have error rates greater than 20%. The histogram and the pie chart further display the accuracy of ANN2.

Table 6-2. ANN2 Average Error

	Average Error
Training Set	1.51%
Validation Set	2.50%
Testing Set	1.99%

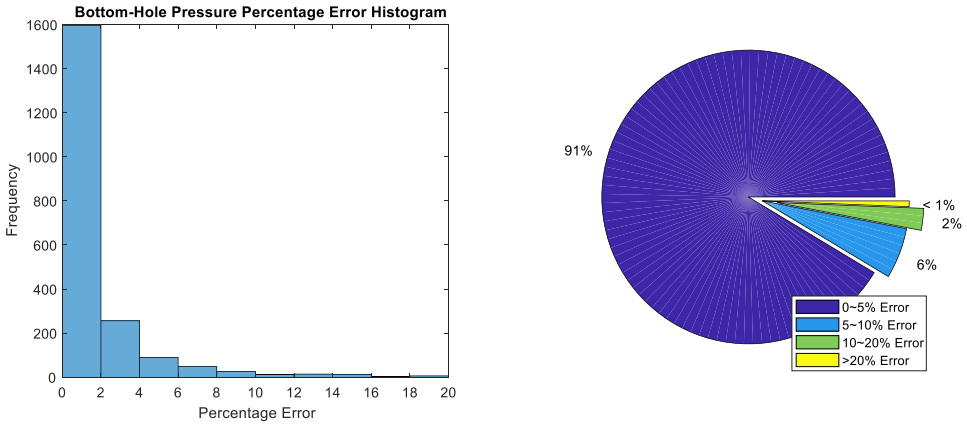


Figure 6-2. ANN2 Bottom-Hole Pressure Error Distribution

Figure 6-3 below shows the best case of ANN2 prediction. ANN predictions precisely match the simulation outputs with single digit variation in bottom-hole pressure. Figure 6-4 is the worst case of ANN2 prediction. As shown in the figure, ANN captures the general trend and magnitude of the output, but the predictions are off by 50-100 psi. The low base values of this case also increase the percentage error rate. It is discovered that this is a rare case because the reservoir is extremely tight with high production rate. The reservoir pressure drops below 1000 psi on the first day of production, and it reaches atomospheric pressure rapidly. The decline curve may not seem smooth mostly due to the magnitutde of the pressure values. Please note that this special case reaches minimal pressure after only nine days. As a result, the two pressure points $P_{wf@day\ 1}$ and $P_{wf@1/6t}$ collapse on the same day due to numerical roundings.

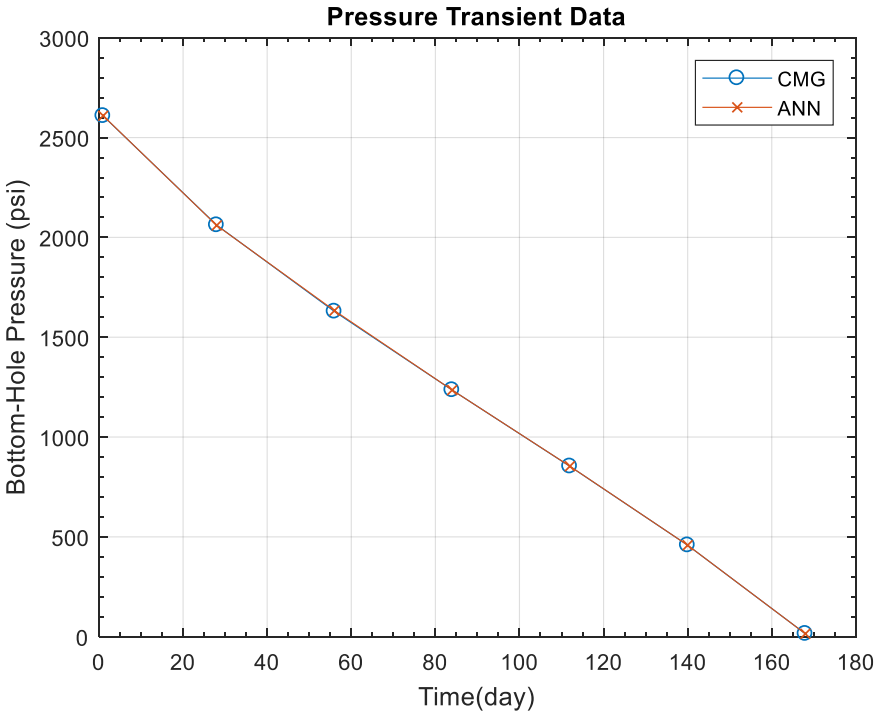


Figure 6-3. Best Case of ANN2 Prediction

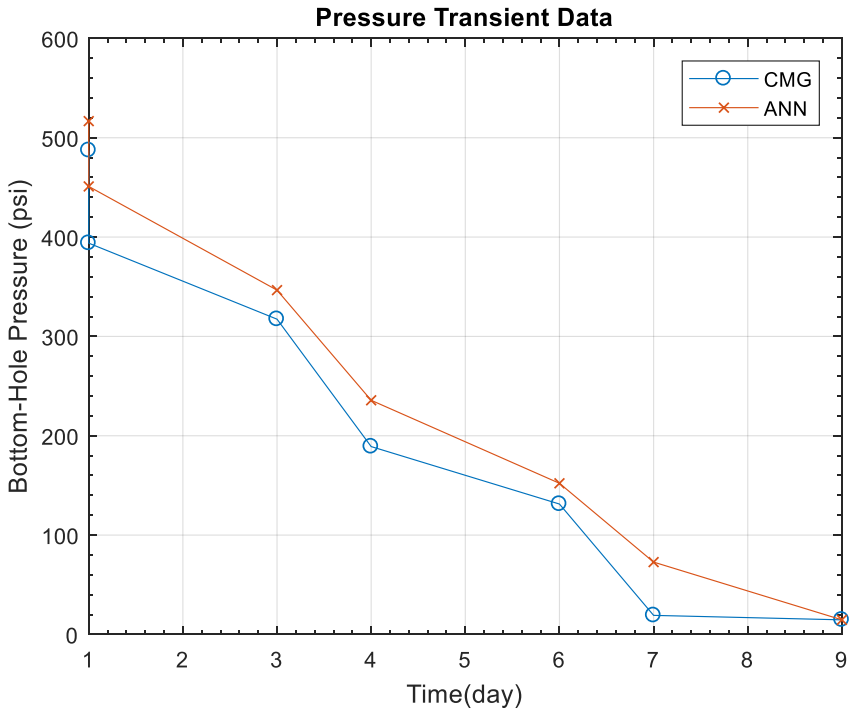


Figure 6-4. Worst Case of ANN2 Prediction

6.2 Inverse ANN Results

The inverse ANN includes ANN3a, ANN3b, ANN4a and ANN4b. Only type II ANNs (ANN4a and ANN4b) will be discussed. Tabulated overall average error and error by parameter are presented in this section. Error distribution in histogram and pie chart for each output parameter is also included. For each ANN, the best and the worst case are shown. In contrast to the accurate forward ANN, inverse ANN displays large error values. The well completion parameters validation tools (ANN3b and ANN4b) are more accurate with all the output parameters under 10% error. On the other hand, the reservoir characteristics prediction tools (ANN3a and ANN4a) can only predict some output parameters within 20% error. Few variables have greater than 50%. Although ANN3a and ANN4a have high error, their solutions are still acceptable after conducting closure tests which are discussed in section 6.3.

6.2.1 ANN3a and ANN4a Results

Although ANN3a and ANN4a have similar performances, ANN3a's prediction is 4%-7% more accurate than ANN4a. The average errors by parameter of two ANNs share the same trend. Results of ANN3a can be found in Appendix D.

Table 6-3 shows the average error of ANN4a for its training, validation and testing set. The performance is less than ideal with all three errors greater than 25%. The most representative testing error is 28% which is much larger than the 20% baseline set for the category of acceptable predictions. Table 6-4 displays the average error of each output parameter within the testing set. It is found that properties such as matrix porosity, SRV fracture porosity, SRV fracture permeability predictions have good accuracy, and their errors are found to be around 10%. Fracture porosity, fracture spacing and SRV fracture spacing have close to acceptable errors at around 25%. Matrix

permeability and fracture permeability have poor performance with average errors greater than 55%.

Table 6-3. ANN4a Average Error

	Average Error
Training Set	26.38%
Validation Set	29.25%
Testing Set	28.03%

Table 6-4. ANN4a Average Error by Parameter

	Average Error
Matrix Porosity (%)	14.12%
Fracture Porosity (%)	22.71%
SRV Fracture Porosity (%)	11.97%
Matrix Permeability (mD)	61.60%
Fracture Permeability (mD)	55.89%
SRV Fracture Permeability (mD)	7.99%
Fracture Spacing (ft)	25.43%
SRV Fracture Spacing (ft)	24.23%

Figure 6-5 below shows the matrix porosity error distribution. Most matrix porosity outputs have error below 10%. The frequency of outputs decreases as the percentage error increases. The pie chart also shows that the errors of the outputs distribute evenly among five ranges. The average error is 14.12%. The error distribution of fracture porosity is illustrated by Figure 6-6. It is discovered that the output errors for fracture porosity are more spread out than the errors of matrix porosity. Only half of the outputs have smaller than 20% error. However, the majority of the high error cases fall into the range of 20%-30% error range. Although there are few cases that have errors as large as 80%, the overall average is set at 22.71%. Figure 6-7 shows the error distribution of SRV fracture porosity. Most cases have errors smaller than 15%. There are only a few outliers

with large errors, and they contribute to 17% of the cases. The overall average error is 11.97% which is satisfactory.

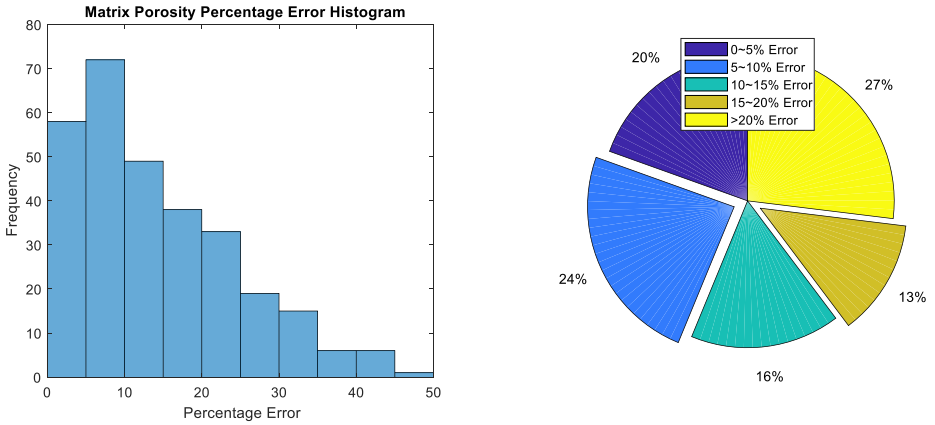


Figure 6-5. ANN4a Matrix Porosity Error Distribution

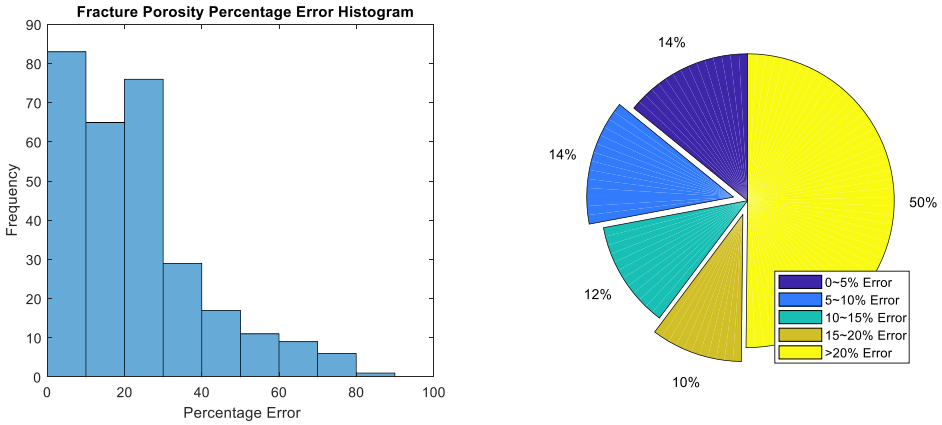


Figure 6-6. ANN4a Fracture Porosity Error Distribution

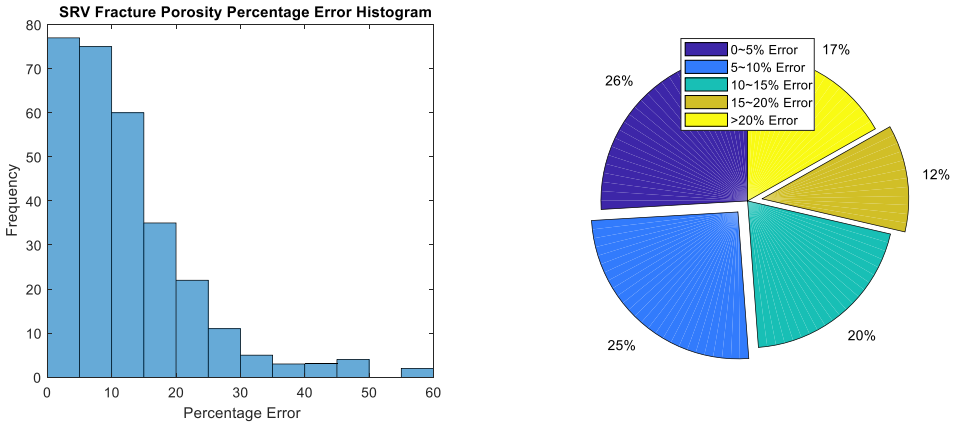


Figure 6-7. ANN4a SRV Fracture Porosity Error Distribution

Figure 6-8 shows the error distribution of matrix permeability. The neural network struggles to estimate this property within tolerable error. Not only 70% of the outputs have errors greater than 20%, but these outputs also have errors as large as 450%. Thus, the average error is 61.60% which is much higher than the acceptable value. The fracture permeability estimation also works poorly as indicated by Figure 6-9. Although more outputs are in the acceptable range, there are more cases with larger than 100% error. The highest error is at around 900%, and the outliers have large impact on increasing the average error which is 55.89%.

Upon further investigation, it is hypothesized that the high errors of some of the outputs are caused by their low base values. For example, matrix permeability is a small number at the magnitude of 1E-05. Small error in prediction is magnified by the small denominator even though the absolute difference is marginal.

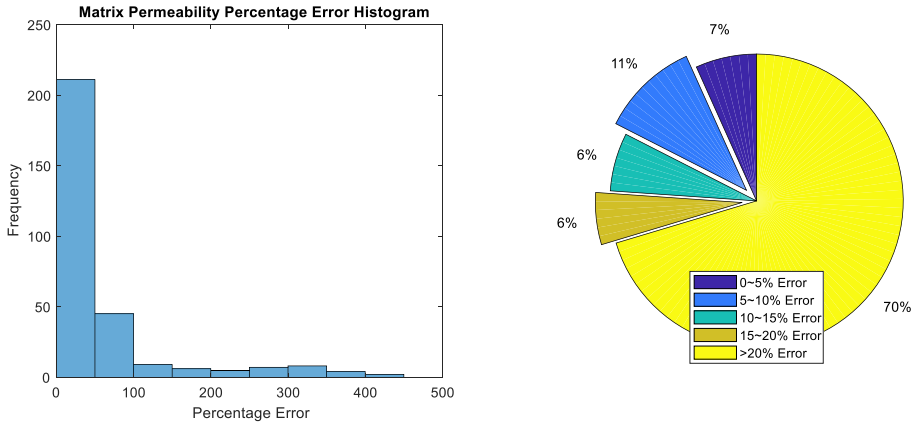


Figure 6-8. ANN4a Matrix Permeability Error Distribution

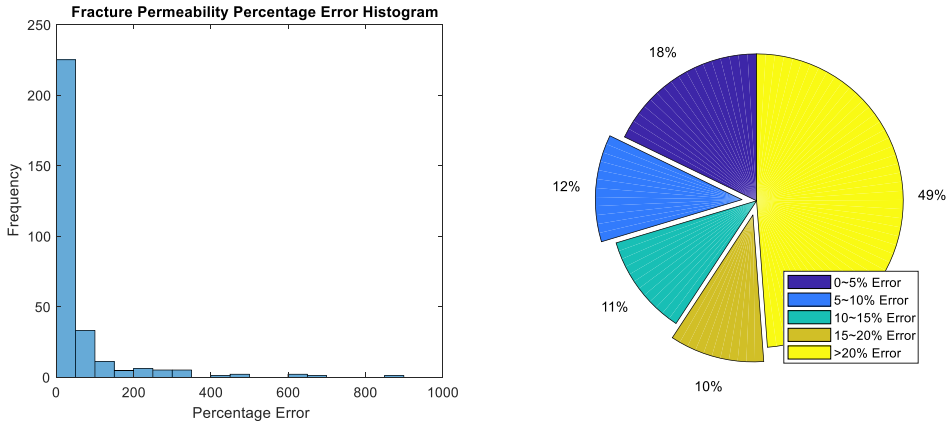


Figure 6-9. ANN4a Fracture Permeability Error Distribution

The error distribution of SRV fracture permeability is shown in Figure 6-10. Nearly half of the outputs have lower than 5% error. The 6% high error cases mostly fall into the range between 20% to 40% error. ANN4a has the best accuracy when predicting SRV fracture permeability and the average error is only 7.99%. Figure 6-11 shows the error distribution of fracture spacing. Although it has 52% of the outputs that have larger than 20% error, the average error is at 25.43%. The accuracy of fracture spacing prediction is slightly lower than 20%. The majority of the cases have smaller than 30% errors with some outliers at 100% error. Figure 6-12 displays the error

distribution of SRV fracture spacing. This error distribution is similar to the distribution of fracture spacing. Despite it has 54% of the outputs that have larger than 20% error, the average is 24.23%.

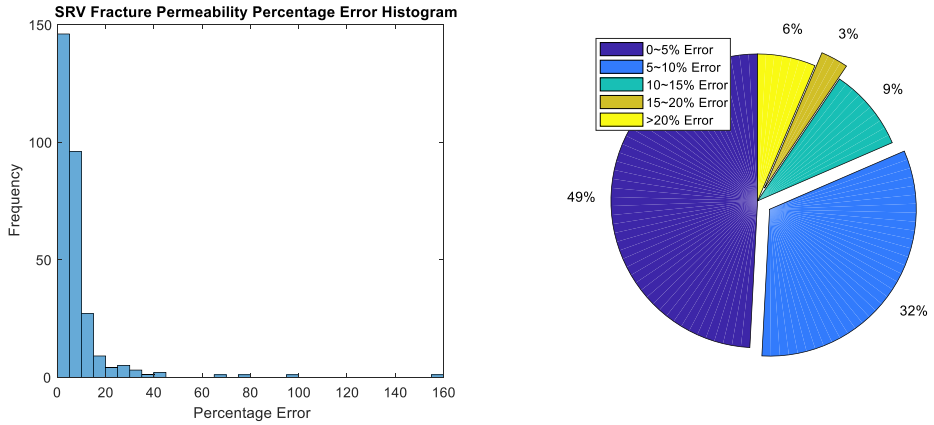


Figure 6-10. ANN4a SRV Fracture Permeability Error Distribution

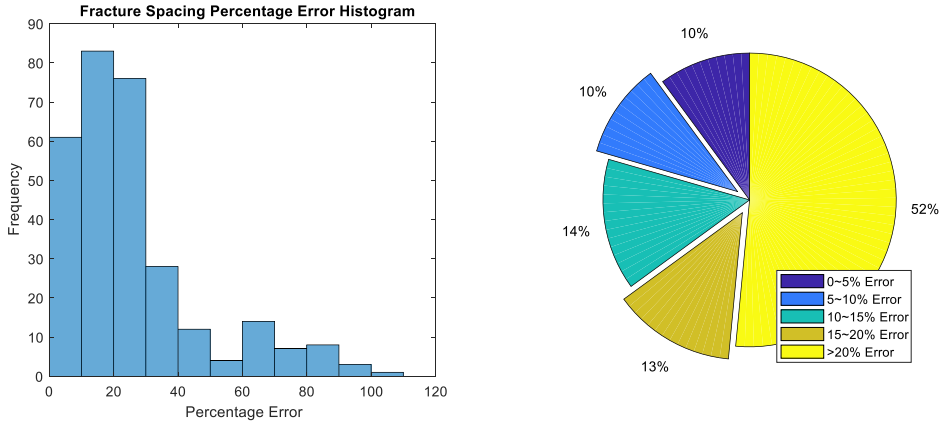


Figure 6-11. ANN4a Fracture Spacing Error Distribution

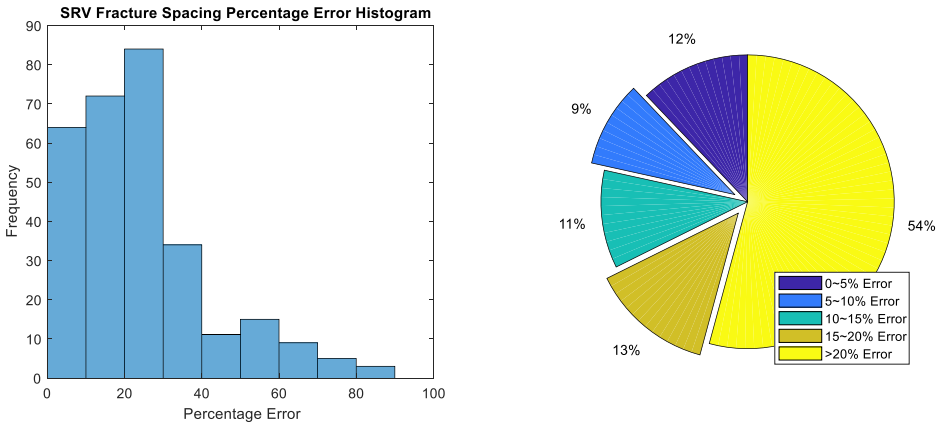


Figure 6-12. ANN4a SRV Fracture Spacing Error Distribution

Table 6-5 and Table 6-6 summarize the best case and the worst case of ANN4a predictions. These two cases are selected based on their average errors. The best case has errors of all parameters but SRV fracture permeability lower than 10%. The worst case has extremely high error in fracture permeability. However, the error of SRV fracture permeability is actually lower than the best case. It can be concluded that the average error is not representative in ANN4a because the error of each parameter various significantly. In addition, the average error is greatly impacted by outliers. As shown in Table 6-6, the high average error is driven by the 851.78% error in fracture permeability.

Table 6-5. Best Case of ANN4a Prediction

	CMG	ANN	Average Error
Matrix Porosity (%)	0.102472	0.094329	7.95%
Fracture Porosity (%)	0.010686	0.009631	9.88%
SRV Fracture Porosity (%)	0.111891	0.119839	7.10%
Matrix Permeability (mD)	6.07E-05	5.73E-05	5.50%
Fracture Permeability (mD)	7.27E-03	0.007123	1.99%
SRV Fracture Permeability (mD)	0.033784	0.029187	13.61%
Fracture Spacing (ft)	2.573273	2.659617	3.36%
SRV Fracture Spacing (ft)	0.266498	0.270835	1.63%

Table 6-6. Worst Case of ANN4a Prediction

	CMG	ANN	Average Error
Matrix Porosity (%)	0.069646	0.09234	32.59%
Fracture Porosity (%)	0.008697	0.009567	9.99%
SRV Fracture Porosity (%)	0.135934	0.114073	16.08%
Matrix Permeability (mD)	6.56E-05	4.59E-05	30.02%
Fracture Permeability (mD)	0.000354	0.003372	851.78%
SRV Fracture Permeability (mD)	0.008753	0.008506	2.82%
Fracture Spacing (ft)	3.860358	2.948325	23.63%
SRV Fracture Spacing (ft)	0.197997	0.278224	40.52%

Despite the large average error of ANN4a, the prediction agrees with the forward ANN in the closure test. The closure test uses the outputs from the inverse ANNs as inputs in the forward ANNs. The method and the results of the closure test is discussed detailly in section 6.3. There are infinite number of combinations of reservoir properties that can generate the same production profile. For example, pressure transient data may not change if the product of permeability and thickness is kept constant. Moreover, the marginal absolute differences in some outputs may not influence the production profile greatly disregarding the high percentage errors.

6.2.2 ANN3b and ANN4b Results

ANN3b and ANN4b have similar output accuracy of well length and flow rate parameters. The error of fracture half-length of ANN4b doubles the error of ANN3b. The average error of ANN4b is around 2% more than ANN3b, and both ANNs have good performance. The detailed results of ANN3b can be found in Appendix D.

Table 6-7 shows the average error of training, validation and testing set of ANN4b. The accuracy is excellent with 5.11% testing error. Table 6-8 displays the average error of ANN4b by parameter. It shows that well length and flow rate outputs have excellent performance with around 3% error. Fracture half-length is slightly less accurate with a 9.14% error margin.

Table 6-7. ANN4b Average Error

	Average Error
Training Set	4.27%
Validation Set	5.51%
Testing Set	5.11%

Table 6-8. ANN4b Average Error by Parameter

	Average Error
Well Length (ft)	2.94%
Flow Rate (mmscf/day)	3.24%
Fracture Half-Length (ft)	9.14%

Figure 6-13 and Figure 6-14 show the error distributions of well length and flow rate outputs. The two parameters have similar histograms and pie charts. For both outputs, more than 80% of the outputs have errors smaller than 5%. The largest error of well length outputs is at 16%, while the flow rate outputs have cases with 37% error. Despite the existence of outliers in flow rate outputs, these two parameters have close average error. Figure 6-15 shows the error distribution of fracture half-length. Less than half of the outputs have error margins smaller than 5%, and 10% of the cases belong to high error category. Outliers with larger than 100% error also exist in the outputs. However, the high frequency of low error cases brings the average error down to 9.14%.

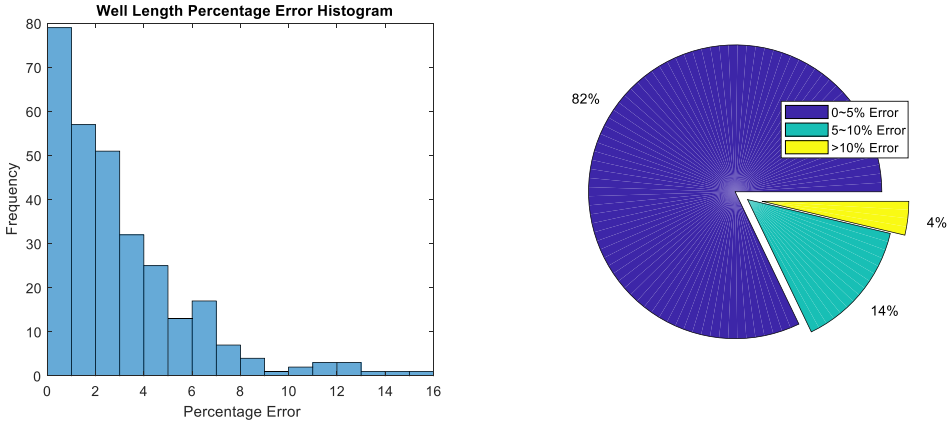


Figure 6-13. ANN4b Well Length Error Distribution

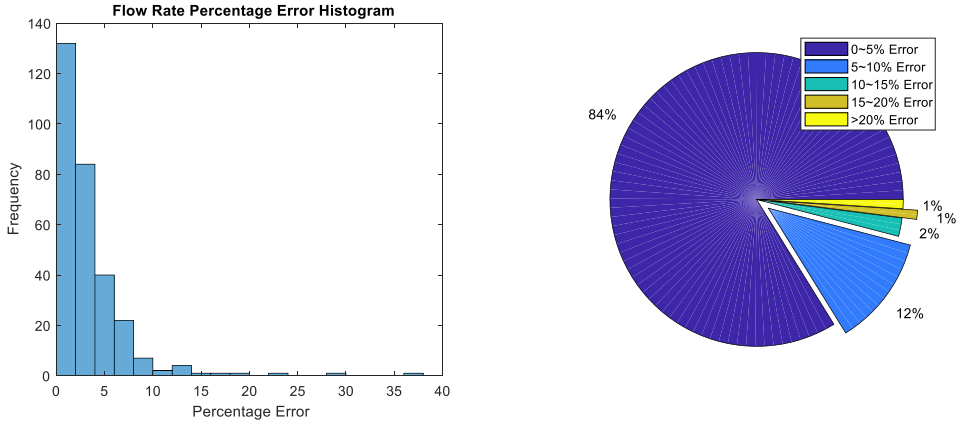


Figure 6-14. ANN4b Flow Rate Error Distribution

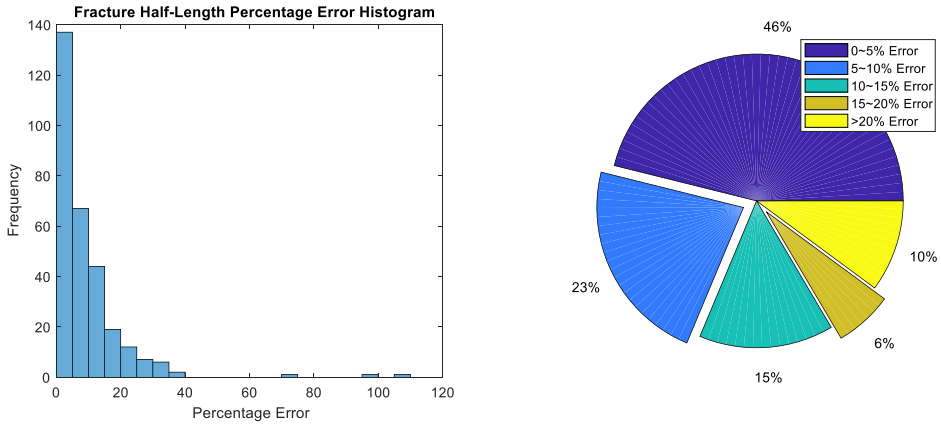


Figure 6-15. ANN4b Fracture Half-Length Error Distribution

Table 6-9 and Table 6-10 show the best and the worst case of ANN4b results. The estimations of the best case exactly match the simulation outputs with smaller than 1% error. The worst case still has good estimations on well length and flow rate but fails to generate fracture half-length accurately. It is discovered that the estimations on well length and flow rate are more robust than the estimations on fracture half-length which has large error outliers. The good performance of the well completion parameters validation tool can decrease the error in the closure test.

Table 6-9. Best Case of ANN4b Prediction

	CMG	ANN	Average Error
Well Length (ft)	3786.165	3789.386	0.09%
Flow Rate (mmscf/day)	8457675	8418950	0.46%
Fracture Half-Length (ft)	193.4317	194.4514	0.53%

Table 6-10. Worst Case of ANN4b Prediction

	CMG	ANN	Average Error
Well Length (ft)	4943.881	4521.139	8.55%
Flow Rate (mmscf/day)	7639150	7923032	3.72%
Fracture Half-Length (ft)	55.89321	116.6413	108.69%

6.3 Closure ANN Test Results

As discussed in section 6.2.1, the objective of the closure test is to prove that the outputs generated by inverse ANNs agree with the forward ANNs. The forward ANNs were used to generate pressure transient data based on the outputs produced by the inverse ANNs. Then, the differences between actual and expected outputs were analyzed. The idea of type II data closure ANN test is illustrated in Figure 6-16. Similar results of type I data can be found in Appendix D. Only the closure test for type II data is discussed to avoid repetition.

20 fresh cases were randomly generated for the test. The data was inputted to ANN0 to estimate the time to reach minimal pressure. All the information was then plugged into ANN2 to generate bottom-hole pressure data. Using the predicted pressure data along with other known properties, ANN4a estimated eight reservoir characteristics including porosity, permeability and fracture spacing. Similarly, PTA data was used in ANN4b to generate well completion parameters. The outputs from the reservoir characteristics prediction tool and the well completion parameter validation tool replaced the corresponding inputs in ANN2. The testing error of the results generated by ANN2 was stored and reported.

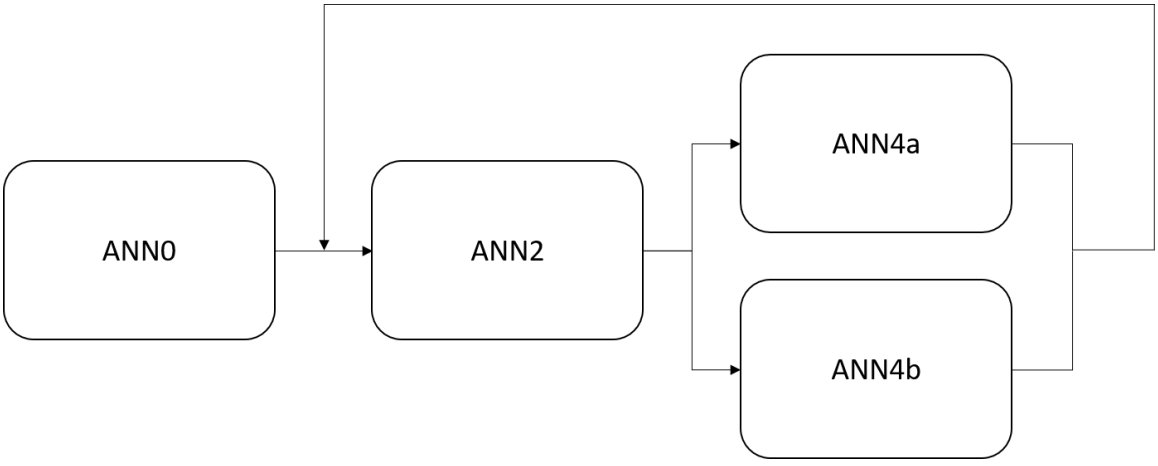


Figure 6-16 Closure Test 2 Diagram

Table 6-11 shows the average error of the closure test for the 20 fresh cases. Despite the large percentage error in ANN4a, the average error of the closure test is only 2.62% which is slightly higher than the 1.99% ANN2 error. The best case is Case 7 with 0.76% error, and the worst case occurs with Case 14 at 6.02% error. Thus, it is safe to conclude that the solution generated by ANN4a is one of the possible close combinations of reservoir characteristics.

Table 6-11. Type II Data Closure Test Average Error of 20 Fresh Cases

Case Number	Average Error
1	2.55%
2	2.15%
3	2.62%
4	3.04%
5	1.68%
6	0.79%
7	0.76%
8	3.60%
9	2.61%
10	3.93%
11	4.43%
12	2.08%
13	4.08%
14	6.02%
15	4.10%
16	2.71%
17	0.95%
18	0.93%
19	0.88%
20	2.46%

Figure 6-17 and Figure 6-18 below show the pressure transient data of the best and the worst cases. The best case has pinpoint accuracy with the predicted pressure data points overlapping the real data points. The worst case has minor variation at early production days, but the accuracy improves as the time increases. The largest absolute difference is about 100 psi.

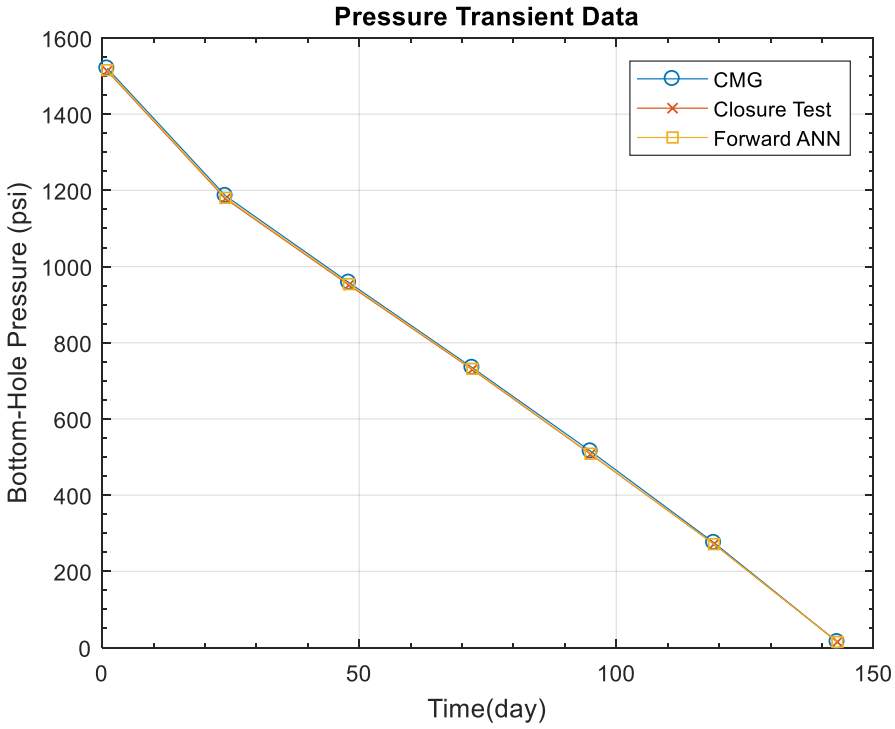


Figure 6-17. Best Case of Type II Data Closure Test Prediction

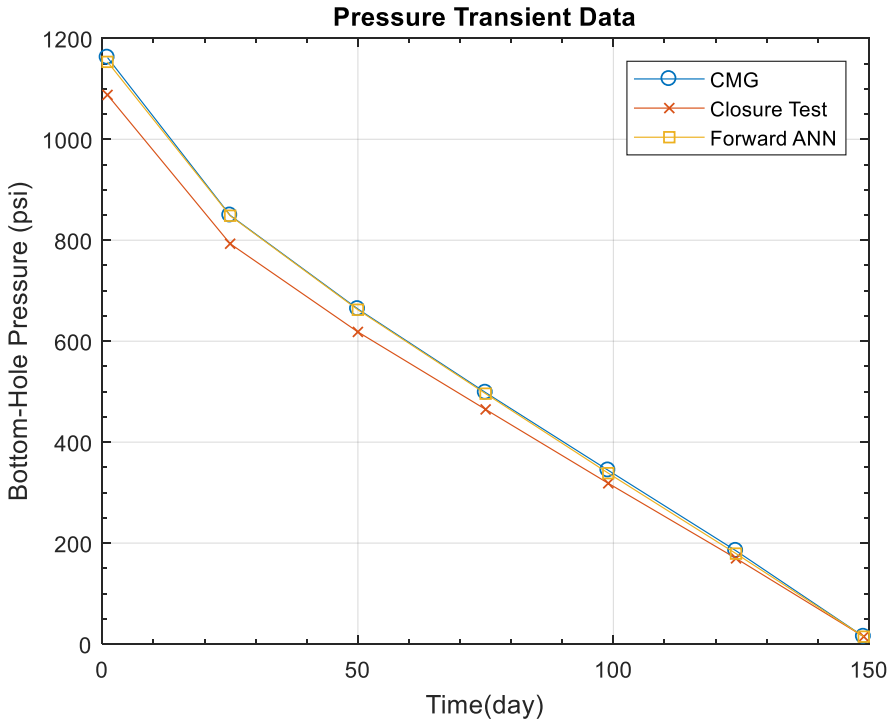


Figure 6-18. Worst Case of Type II Data Closure Test Prediction

Chapter 7

Development of Graphical User Interface

After all the ANNs have been verified and tested, it is important to create a simple graphical user interface (GUI). Especially for the users who are not familiar with either MATLAB or artificial neural networks, the GUI allows the user to generate desired results without having strong understandings of the materials. The GUI also discourages the user to edit MATLAB data or script. Editing backend files can create potential errors and drive the expert system to incorrect results.

All five GUIs are created using MATLAB GUI Toolbox. Figure 7-1 displays the start menu of the graphical user interface. The start menu is the home panel for all four GUIs created for the expert system. Each button includes the names of the associated ANNs and their functions. User simply need to click on the button to navigate to the correct tool.

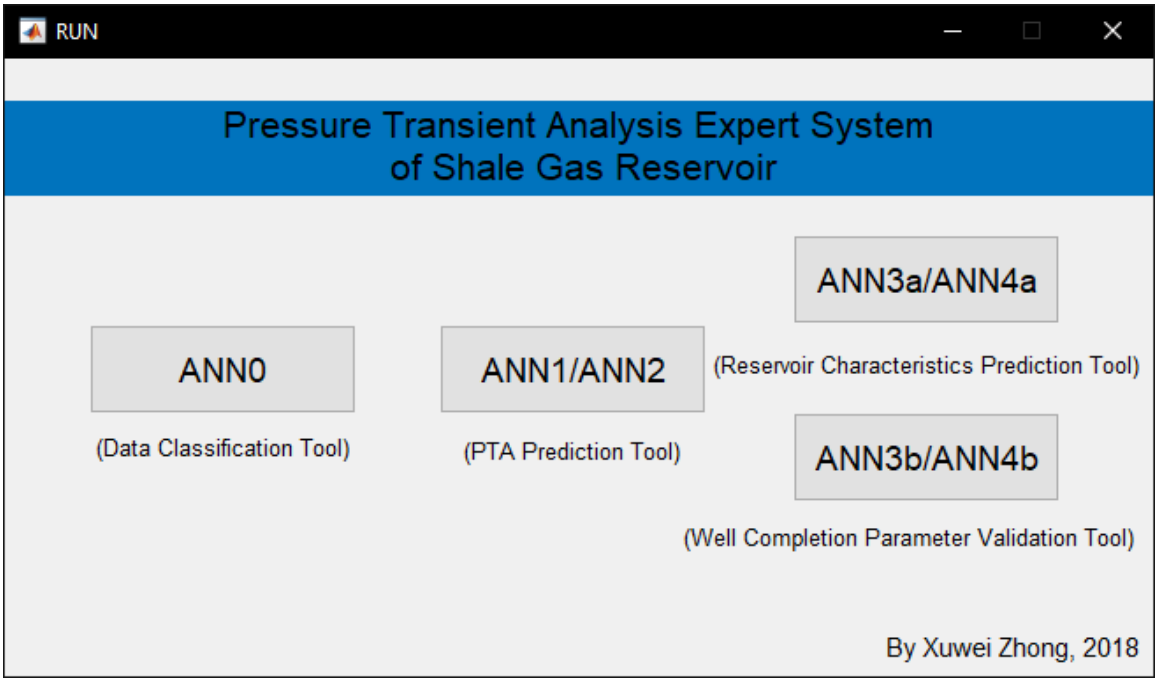


Figure 7-1. Start Menu Graphical User Interface

7.1 Graphical User Interface for Forward ANN

Two graphical user interfaces are created for forward ANN. Figure 7-2 shows the GUI of data type classification tool. 15 parameters are required for inputs. The first two columns of the input table include minimum and maximum values. The third column is editable user's inputs. The user should enter a reasonable value within the given range. After completing the input table, the user can click the "Predict with ANN0" button to continue. Predicted data type and time to reach minimum pressure will then display in the output area. The user can return to home panel by clicking "Back" button. The "Save Data" button saves user's input and generated output in a separate data file which can be accessed by other GUIs.

Figure 7-3 demonstrates the pressure transient data prediction tool GUI. The input table is similar to the one in Figure 7-2 with an additional row for time to reach minimum pressure. Note that ANN1 does not require nor use this additional parameter. An output table and a remarks area are located below the input table. The remarks area is used to define the variable t used in the output table. The input table can be filled by either user's input or loading stored data. The "Load Data" button copies the stored data to the input table. The user can predict pressure transient data with either ANN1 or ANN2 by clicking the correct button. After the internal calculations are completed, the output table will be automatically filled. Additionally, a small figure that plots the seven bottom-hole pressure points versus time will appear in the bottom-right corner. Similar to ANN0, the "Save Data" button stores the updated data which includes both input and output tables.

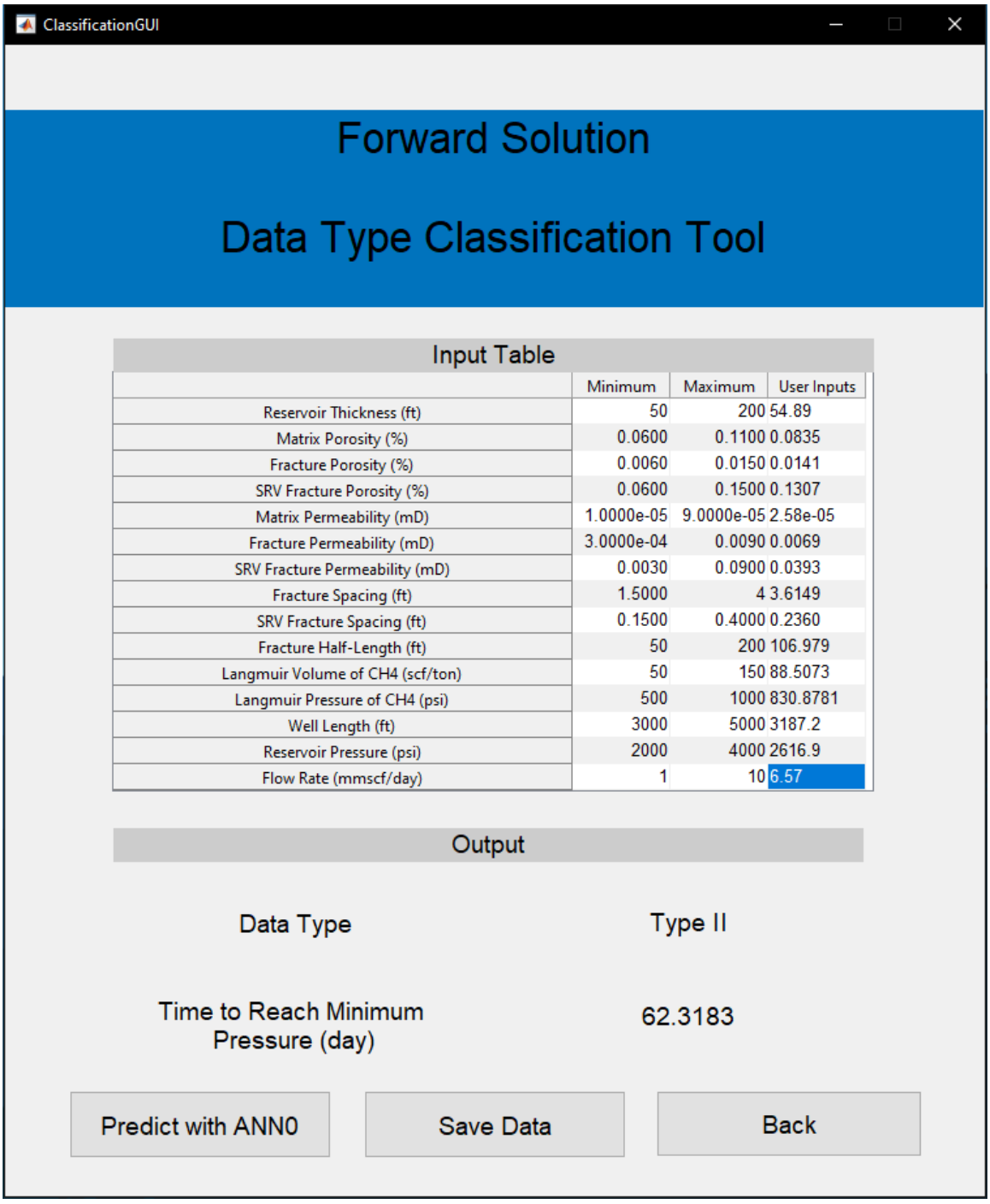


Figure 7-2. Data Type Classification Tool Graphical User Interface

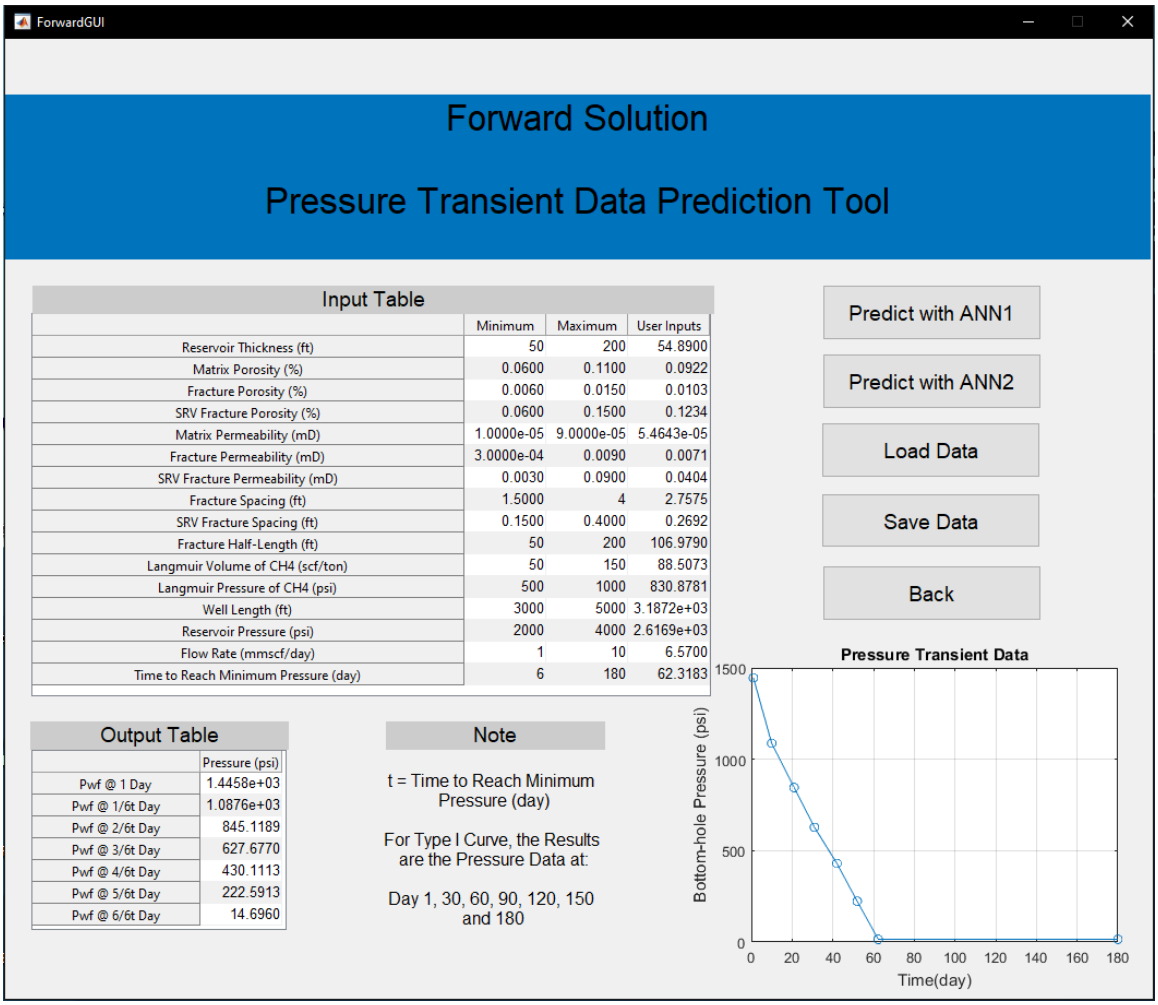


Figure 7-3. Pressure Transient Data Prediction Tool Graphical User Interface

7.2 Graphical User Interface for Inverse ANN

Similar design is used in inverse solution GUIs. Figure 7-4 shows the user interface of reservoir characteristics prediction tool. The input table includes seven pressure points along with selected reservoir and well completion parameters. The output table has eight reservoir characteristics outputs that will automatically update after user's action. The GUI allows the user to use either ANN3a or ANN4a to predict reservoir properties.

Figure 7-5 displays the GUI of well completion parameter validation tool. The input table is expanded to 20 inputs. Well length, flow rate and fracture half-length are updated based on user's inputs. ANN3a and ANN4a predictions are both available to use at user's demand.

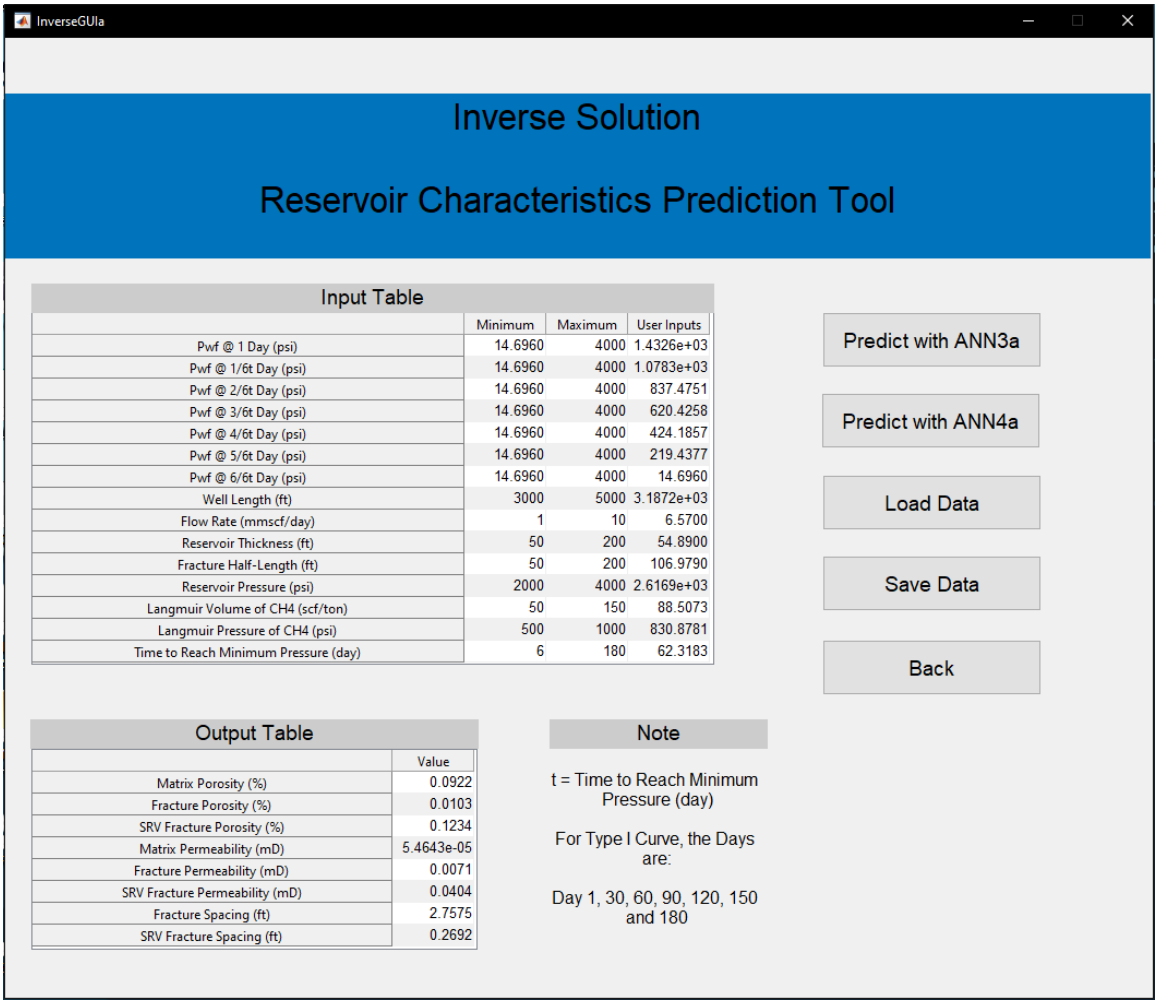


Figure 7-4. Reservoir Characteristics Prediction Tool Graphical User Interface

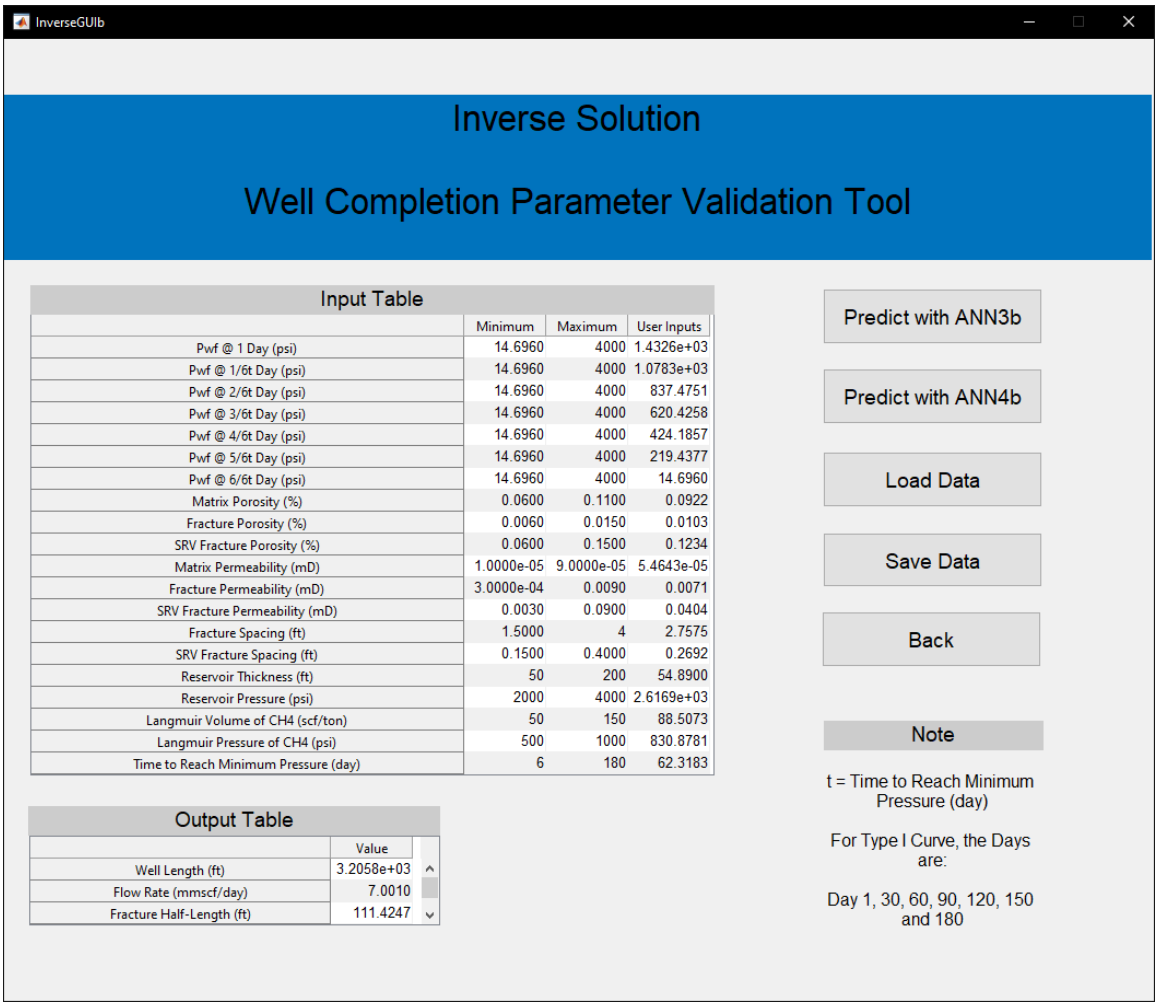


Figure 7-5. Well Completion Parameter Prediction Tool Graphical User Interface

Chapter 8

Conclusions

In conclusion, the goal of this study is to develop an expert system that utilizes artificial neural network to analyze pressure transient data in shale reservoirs. CMG-GEM software, MATLAB Neural Network Toolbox and MATLAB GUI Toolbox are used in the research. A simulation model with horizontal well is developed for pressure transient data generation purposes. In total, seven ANNs with different functions are produced. ANN0, ANN1 and ANN2 are constructed as forward ANNs that can predict pressure transient data based on reservoir and well completion parameters. Inverse ANN is made of ANN3a, ANN3b, ANN4a and ANN4b. ANN3a and ANN4a comprise a reservoir characteristics prediction tool that can estimate important reservoir properties using production data and well completion parameters. ANN3b and ANN4b form a well completion parameter validation tool that closes the loop of closure test. Moreover, five graphical user interfaces (GUI) are developed for the ease of use by future users.

The following conclusions are drawn from this study:

1. Forward ANN is capable of predicting pressure transient data accurately with less than 5% average error.
2. The reservoir characteristics prediction tool of inverse ANN has greater than 20% average error. The accuracy is less than ideal, however, analyzed results show that during the closure checks the predicted values in combined generate successful agreements with the original forward problem.

3. The well completion parameter validation tool of inverse ANN is accurate with lower than 5% average error.
4. According to the closure test results, reservoir characteristics prediction tool generates correct solution for the reservoir model despite having large average error.
5. Training of an ANN is computationally expensive. However, once the network has been correctly trained and validated, the results can be generated within seconds.
6. To prevent overfitting, the complexity of the architecture of an ANN needs to be determined according to the size of data.
7. Adding more layers and neurons to an ANN does not necessary improves its performance. In some cases, the performance will decrease.
8. Use of functional links does not improve the accuracy of ANNs significantly in this study. Thus, the functional links are excluded from the final design.
9. The quality of the data has the greatest influence on the accuracy of an ANN. Architecture of the ANN has the second largest impact on the performance.
10. Sigmoid transfer functions are found to provide the best results when used between hidden layers. Hyperbolic tangent sigmoid transfer function should be used to suit the data ranging from -1 to 1. For data ranging from 0 to 1, log-sigmoid transfer function should be implemented.
11. ANN with large number of data sets has more stable performance than an ANN with small number of data sets. The accuracy of the ANN with small data set is more influenced by the random division of the data into training, validation and testing sets.

Suggestions for future work:

1. Generating large number of simulation cases is time and space expensive. Alternately, the forward solution can be trained and tested with smaller number of cases first. If the forward solution predicts accurately, it can be used to generate more simulation cases. This approach should save computational time and space. Also, it creates the possibility of generating a much larger training data set, which helps to make the neural network more robust.
2. More data screening criteria can be implemented. Pressure transient curves that have zigzag behavior should be eliminated from the training set.
3. Additional wellbore structures can be studied and analyzed.
4. Real reservoir data can be used to test the expert system.
5. Anisotropic reservoirs can be studied. Since it involves introducing more inputs and outputs, a more advanced machine learning method may be required.
6. Rock compressibility terms and skin factor of the well can be included.
7. The neural network approach of analyzing well testing data can be extended to other unconventional reservoirs and multi-phase reservoirs.

References

Alajmi, M., & Ertekin, T. (2007). The Development of an Artificial Neural Network as a Pressure Transient Analysis Tool for Applications in Double-Porosity Reservoirs. *Proceedings of Asia Pacific Oil and Gas Conference and Exhibition*. doi:10.2523/108604-ms

Alqahtani, M. H., & Ertekin, T. (2017). Shale Gas Reservoir Development Strategies using Complex Specified Bottom-Hole Pressure Well Architectures. *SPE Kingdom of Saudi Arabia Annual Technical Symposium and Exhibition*. doi:10.2118/188144-ms

Aptekar-Cassels, W. (2017, March 22). Neural Networks [Web log post]. Retrieved from http://blog.wesleyac.com/posts/visualizing-neural-network-learning

Aydinoglu, G., Bhat, M., & Ertekin, T. (2002). Characterization of Partially Sealing Faults from Pressure Transient Data: An Artificial Neural Network Approach. *Proceedings of SPE Eastern Regional Meeting*. doi:10.2523/78715-ms

Beale, M. H., Hagan, M. T., & Demuth, H. B. (2018, March). *Neural Network Toolbox User's Guide*. MathWorks.

Beckwith, R. (2013). The Marcellus Shale Gas Boom Evolves. *Journal of Petroleum Technology,65*(06), 34-44. doi:10.2118/0613-0034-jpt

Belvalkar, R. A., & Oyewole, S. (2010). Development of Marcellus Shale in Pennsylvania. *SPE Annual Technical Conference and Exhibition*. doi:10.2118/134852-ms

Buscema, M. (2009, July 3). A Brief Overview and Introduction to Artificial Neural Netoworks. *Substance Use & Misuse,37*(8-10), 1093-1148.

Centilmen, A., Ertekin, T., & Grader, A. (1999). Applications of Neural Networks in Multiwell Field Development. *Proceedings of SPE Annual Technical Conference and Exhibition*. doi:10.2523/56433-ms

Clarkson, C. R., & Haghshenas, B. (2013). Modeling of Supercritical Fluid Adsorption on Organic-Rich Shales and Coal. *SPE Unconventional Resources Conference-USA*. doi:10.2118/164532-ms

Cox, J. S. (2014). *The Development of an Artificial Neural Network as a Pressure Transient Analysis Tool with Applications in Multi-Lateral Wells in Tight-Gas Dual-Porosity Reservoirs*. The Pennsylvania State University.

Gaw, H. (2014). *Development of an Artificial Neural Network for Pressure and Rate Transient Analysis of Horizontal Wells Completed in Dry, Wet and Condensate Gas Reservoirs of Naturally Fractured Formations*. The Pennsylvania State University.

Gevin, G. (1997). *An introduction to neural networks*. UCL Press.

Hagan, M. T., Demuth, H. B., & Beale, M. H. (1996). *Neural network design*. Boston: PWS Publishing Company.

Heaton, J. (2008). *Introduction to Neural Networks for Java*.

Kuchuk, F. J., Goode, P. A., Brice, B. W., Sherrard, D. W., & Thambynayagam, R. M. (1990). Pressure-Transient Analysis for Horizontal Wells. *Journal of Petroleum Technology*,*42*(08), 974-1031. doi:10.2118/18300-pa

Kulga, I. B. (2014). *Analysis of the Efficacy of Carbon Dioxide Sequestration in Depleted Shale Gas Reservoirs*. The Pennsylvania State University.

Langmuir, I. (1918). The Adsorption of Gases on Plane Surfaces Of Glass, Mica And Platinum. *Journal of the American Chemical Society*,*40*(9), 1361-1403. doi:10.1021/ja02242a004

Liu, L. (2017). *Analysis of the Efficacy of Carbon Dioxide Sequestration in Depleted Coalbed Methane Reservoirs*. The Pennsylvania State University.

Lu, J. (2015). *Rate Transient Analysis of Dual Lateral Wells in Naturally Fractured Reservoirs via Artificial Intelligence*. The Pennsylvania State University.

MATLAB App Building. (2018, March). MathWorks.

Møller, M. F. (1990). A Scaled Conjugate Gradient Algorithm for Fast Supervised Learning. *DAIMI Report Series,19*(339). doi:10.7146/dpb.v19i339.6570

Priddy, K. L., & Keller, P. E. (n.d.). Data Normalization. *Artificial Neural Networks: An Introduction*. doi:10.1117/3.633187.ch3

Qiao, J., Li, F., Han, H., & Li, W. (2016). Constructive algorithm for fully connected cascade feedforward neural networks. *Neurocomputing,182*, 154-164. doi:10.1016/j.neucom.2015.12.003

Sapag, K., Vallone, A., Garca, A., & Solar, C. (2010). Adsorption of Methane in Porous Materials as the Basis for the Storage of Natural Gas. *Natural Gas*. doi:10.5772/9846

Shang, B. (2017). *Design of Brine Disposal Wells in Depleted Gas Reservoirs via Artificial Intelligence Protocols*. The Pennsylvania State University.

Walton, I., & Mcle, J. (2013). The Role of Natural Fractures in Shale Gas Production. *Effective and Sustainable Hydraulic Fracturing*. doi:10.5772/56404

Warren, J., & Root, P. (1963). The Behavior of Naturally Fractured Reservoirs. *Society of Petroleum Engineers Journal,3*(03), 245-255. doi:10.2118/426-pa

Where Our Natural Gas Comes From. (2017, October 25). Retrieved from https://www.eia.gov/energyexplained/index.php?page=natural_gas_where

Woodford, C. (2016, March 18). How Neural Networks Word. Retrieved from https://www.explainthatstuff.com/introduction-to-neural-networks.html

Yu, W., Sepehrnoori, K., & Patzek, T. W. (2014). Evaluation of Gas Adsorption in Marcellus Shale. *SPE Annual Technical Conference and Exhibition*. doi:10.2118/170801-ms

Yu, W., Sepehrnoori, K., & Patzek, T. W. (2016). Modeling Gas Adsorption in Marcellus Shale With Langmuir and BET Isotherms. *SPE Journal,21*(02), 589-600. doi:10.2118/170801-pa

Zhang, J. (2017). *Development of Automated Neuro-Simulation Protocols for Pressure and Rate Transient Analysis Applications*. The Pennsylvania State University.

Zhou, Q. (2013). *Development and Application of an Artificial Expert System for the Pressure Transient Analysis of a Dual-Lateral Well Configuration*. The Pennsylvania State University.

Appendix A

CMG-GEM DAT File Template

```
RESULTS SIMULATOR GEM 201410

INUNIT FIELD
WSRF WELL 1
WSRF GRID TIME
OUTSRF GRID SO SG SW PRES
OUTSRF RES NONE
WPRN GRID 0
OUTPRN GRID NONE
OUTPRN RES NONE

**  Distance units: ft
RESULTS XOFFSET         0.0000
RESULTS YOFFSET         0.0000

**  (DEGREES)
**  (DEGREES)
RESULTS ROTATION         0.0000  **  (DEGREES)
RESULTS AXES-DIRECTIONS 1.0 -1.0 1.0

**
************************************************************************
** Definition of fundamental cartesian grid
**
************************************************************************
GRID VARI 37 37 1
KDIR DOWN
DI IVAR
 %%LENGTHC%% 3*%%LENGTHB%% 29*%%LENGTHA%%
3*%%LENGTHB%% %%LENGTHC%%
DJ JVAR
 5*%%WIDTHC%% 11*%%WIDTHB%% 5*%%WIDTHA%% 11*%%WIDTHB%%
5*%%WIDTHC%%
DK ALL
 1369*%%THICKNESS%%
DTOP
 1369*6500
DUALPOR
SHAPE WR
**  0 = null block, 1 = active block
NULL *MATRIX CON          1
**  0 = null block, 1 = active block
NULL *FRACTURE CON          1
```

```
        POR *MATRIX CON        %%MATPOR%%
        POR *FRACTURE ALL
         598*%%FRCPOR%% 25*%%SRVFRCPOR%% 11*%%FRCPOR%%
27*%%SRVFRCPOR%% 9*%%FRCPOR%% 29*%%SRVFRCPOR%% 9*%%FRCPOR%%
27*%%SRVFRCPOR%%
         11*%%FRCPOR%% 25*%%SRVFRCPOR%% 598*%%FRCPOR%%
        PERMI *FRACTURE ALL
         598*%%FRCPERM%% 25*%%SRVFRCPERM%% 11*%%FRCPERM%%
27*%%SRVFRCPERM%% 9*%%FRCPERM%% 29*%%SRVFRCPERM%%
9*%%FRCPERM%% 27*%%SRVFRCPERM%%
         11*%%FRCPERM%% 25*%%SRVFRCPERM%% 598*%%FRCPERM%%
        PERMI *MATRIX CON      %%MATPERM%%
        PERMJ *MATRIX CON      %%MATPERM%%
        PERMJ *FRACTURE ALL
         598*%%FRCPERM%% 25*%%SRVFRCPERM%% 11*%%FRCPERM%%
27*%%SRVFRCPERM%% 9*%%FRCPERM%% 29*%%SRVFRCPERM%%
9*%%FRCPERM%% 27*%%SRVFRCPERM%%
         11*%%FRCPERM%% 25*%%SRVFRCPERM%% 598*%%FRCPERM%%
        PERMK *FRACTURE ALL
         598*%%FRCPERM%% 25*%%SRVFRCPERM%% 11*%%FRCPERM%%
27*%%SRVFRCPERM%% 9*%%FRCPERM%% 29*%%SRVFRCPERM%%
9*%%FRCPERM%% 27*%%SRVFRCPERM%%
         11*%%FRCPERM%% 25*%%SRVFRCPERM%% 598*%%FRCPERM%%
        PERMK *MATRIX CON      %%MATPERM%%
        DJFRAC ALL
         598*%%FRCSPAC%% 25*%%SRVFRCSPAC%% 11*%%FRCSPAC%%
27*%%SRVFRCSPAC%% 9*%%FRCSPAC%% 29*%%SRVFRCSPAC%%
9*%%FRCSPAC%% 27*%%SRVFRCSPAC%% 11*%%FRCSPAC%%
         25*%%SRVFRCSPAC%% 598*%%FRCSPAC%%
        DIFRAC ALL
         598*%%FRCSPAC%% 25*%%SRVFRCSPAC%% 11*%%FRCSPAC%%
27*%%SRVFRCSPAC%% 9*%%FRCSPAC%% 29*%%SRVFRCSPAC%%
9*%%FRCSPAC%% 27*%%SRVFRCSPAC%% 11*%%FRCSPAC%%
         25*%%SRVFRCSPAC%% 598*%%FRCSPAC%%
        DKFRAC ALL
         598*%%FRCSPAC%% 25*%%SRVFRCSPAC%% 11*%%FRCSPAC%%
27*%%SRVFRCSPAC%% 9*%%FRCSPAC%% 29*%%SRVFRCSPAC%%
9*%%FRCSPAC%% 27*%%SRVFRCSPAC%% 11*%%FRCSPAC%%
         25*%%SRVFRCSPAC%% 598*%%FRCSPAC%%
        **  0 = pinched block, 1 = active block
        PINCHOUTARRAY CON          1
        SECTORARRAY 'SRV' MATRIX  ALL
         598*0 25*1 11*0 27*1 9*0 29*1 9*0 27*1 11*0 25*1 598*0
        SECTORARRAY 'SRV' FRACTURE  ALL
         598*0 25*1 11*0 27*1 9*0 29*1 9*0 27*1 11*0 25*1 598*0
        CPOR FRACTURE 1e-5
        CPOR MATRIX 1e-5
        **The following is the fluid component
        **property data in GEM format.
```

```
**The unit system and fluid compositions should
**be specified in the I/O control section.
**The units and compositions specified in WinProp
**are included here as comments for informational purposes.
** PVT UNITS CONSISTENT WITH *INUNIT *FIELD
**COMPOSITION *PRIMARY
**         1.0000000E+00
**COMPOSITION *SECOND
**         0.0000000E+00
** Model and number of components
MODEL PR
NC 1 1
COMPNAME 'CH4'
TRES 142
VISCOR HZYT
MIXVC 1.0000000E+00
MW
1.6043000E+01
AC
0.008
PCRIT
4.5400000E+01
VCRIT
9.9000000E-02
TCRIT
1.9060000E+02
PCHOR
77
SG
0.3
TB
-258.61
HCFLAG
0
OMEGA
0.457236
OMEGB
0.0777961
VSHIFT
0
VISVC
9.9000000E-02
DENW 62.4779
ROCKFLUID
RPT 1
**       Sw      krw     krow
**       Sw      krw     krow     Pcow
SWT
       0.22       0       1       7
```

```
        0.3     0.08     0.41        4
        0.4     0.17    0.128        3
        0.5     0.26   0.0645      2.5
        0.6     0.35   0.0045        2
        0.8     0.68        0        1
        0.9     0.85        0      0.5
          1        1        0        0
**       Sl      krg     krog
**       Sg      krg     krog     Pcog
SGT
          0        0        1        0
       0.04    0.005    0.602     0.21
        0.1    0.022    0.333     0.55
        0.2      0.1    0.104     1.03
        0.3     0.24    0.021     1.54
        0.4     0.34        0     2.09
        0.5     0.42        0     2.51
        0.6      0.5        0     3.05
        0.7   0.8125        0      3.5
       0.78        1        0     3.93

ADSORBTMAX 'CH4' %%ADSCH4MAX%%

**  Partial Pressure   Adsorption
**  Partial Pressure   Adsorption
**  Partial Pressure  Adsorption
**  Partial Pressure   Adsorption
ADSTAB 'CH4'
**  Partial Pressure   Adsorption
                                    %%ADSCH4%%
RPT 2
**       Sw      krw     krow
SWT
         0        0        1
         1        1        0
**       Sg      krg     krog
SGT
       0.01        0        1
         1        1        0
INITIAL

USER_INPUT
PRES *MATRIX CON        %%PRESSURE%%
PRES *FRACTURE CON        %%PRESSURE%%
SW *MATRIX CON         0.1
SW *FRACTURE CON       0.001
ZGLOBALC 'CH4' *FRACTURE CON          1
ZGLOBALC 'CH4' *MATRIX CON          1
NUMERICAL
```

```
DTMAX 1
DTMIN 1e-8
RUN
DATE 2017 1 1
DTWELL 1e-005
**
WELL  'Well-1'
PRODUCER 'Well-1'
OPERATE  MAX  STG  %%FLOWRATE%%  CONT
**         rad  geofac  wfrac  skin
GEOMETRY  K  0.25  0.37  1.0  0.0
     PERF      GEOA  'Well-1'
** UBA              ff         Status  Connection
   5 19 1         1.0  OPEN    FLOW-TO  'SURFACE'  REFLAYER
   6 19 1         1.0  OPEN    FLOW-TO  1
   7 19 1         1.0  OPEN    FLOW-TO  2
   8 19 1         1.0  OPEN    FLOW-TO  3
   9 19 1         1.0  OPEN    FLOW-TO  4
   10 19 1        1.0  OPEN    FLOW-TO  5
   11 19 1        1.0  OPEN    FLOW-TO  6
   12 19 1        1.0  OPEN    FLOW-TO  7
   13 19 1        1.0  OPEN    FLOW-TO  8
   14 19 1        1.0  OPEN    FLOW-TO  9
   15 19 1        1.0  OPEN    FLOW-TO  10
   16 19 1        1.0  OPEN    FLOW-TO  11
   17 19 1        1.0  OPEN    FLOW-TO  12
   18 19 1        1.0  OPEN    FLOW-TO  13
   19 19 1        1.0  OPEN    FLOW-TO  14
   20 19 1        1.0  OPEN    FLOW-TO  15
   21 19 1        1.0  OPEN    FLOW-TO  16
   22 19 1        1.0  OPEN    FLOW-TO  17
   23 19 1        1.0  OPEN    FLOW-TO  18
   24 19 1        1.0  OPEN    FLOW-TO  19
   25 19 1        1.0  OPEN    FLOW-TO  20
   26 19 1        1.0  OPEN    FLOW-TO  21
   27 19 1        1.0  OPEN    FLOW-TO  22
   28 19 1        1.0  OPEN    FLOW-TO  23
   29 19 1        1.0  OPEN    FLOW-TO  24
   30 19 1        1.0  OPEN    FLOW-TO  25
   31 19 1        1.0  OPEN    FLOW-TO  26
   32 19 1        1.0  OPEN    FLOW-TO  27
   33 19 1        1.0  OPEN    FLOW-TO  28
DATE 2017 1  2.00000
DATE 2017 1  3.00000
DATE 2017 1  4.00000
DATE 2017 1  5.00000
DATE 2017 1  6.00000
DATE 2017 1  7.00000
DATE 2017 1  8.00000
```

DATE 2017 1 9.00000
DATE 2017 1 10.00000
DATE 2017 1 11.00000
DATE 2017 1 12.00000
DATE 2017 1 13.00000
DATE 2017 1 14.00000
DATE 2017 1 15.00000
DATE 2017 1 16.00000
DATE 2017 1 17.00000
DATE 2017 1 18.00000
DATE 2017 1 19.00000
DATE 2017 1 20.00000
DATE 2017 1 21.00000
DATE 2017 1 22.00000
DATE 2017 1 23.00000
DATE 2017 1 24.00000
DATE 2017 1 25.00000
DATE 2017 1 26.00000
DATE 2017 1 27.00000
DATE 2017 1 28.00000
DATE 2017 1 29.00000
DATE 2017 1 30.00000
DATE 2017 1 31.00000
DATE 2017 2 1.00000
DATE 2017 2 2.00000
DATE 2017 2 3.00000
DATE 2017 2 4.00000
DATE 2017 2 5.00000
DATE 2017 2 6.00000
DATE 2017 2 7.00000
DATE 2017 2 8.00000
DATE 2017 2 9.00000
DATE 2017 2 10.00000
DATE 2017 2 11.00000
DATE 2017 2 12.00000
DATE 2017 2 13.00000
DATE 2017 2 14.00000
DATE 2017 2 15.00000
DATE 2017 2 16.00000
DATE 2017 2 17.00000
DATE 2017 2 18.00000
DATE 2017 2 19.00000
DATE 2017 2 20.00000
DATE 2017 2 21.00000
DATE 2017 2 22.00000
DATE 2017 2 23.00000
DATE 2017 2 24.00000
DATE 2017 2 25.00000
DATE 2017 2 26.00000

DATE 2017 2 27.00000
DATE 2017 2 28.00000
DATE 2017 3 1.00000
DATE 2017 3 2.00000
DATE 2017 3 3.00000
DATE 2017 3 4.00000
DATE 2017 3 5.00000
DATE 2017 3 6.00000
DATE 2017 3 7.00000
DATE 2017 3 8.00000
DATE 2017 3 9.00000
DATE 2017 3 10.00000
DATE 2017 3 11.00000
DATE 2017 3 12.00000
DATE 2017 3 13.00000
DATE 2017 3 14.00000
DATE 2017 3 15.00000
DATE 2017 3 16.00000
DATE 2017 3 17.00000
DATE 2017 3 18.00000
DATE 2017 3 19.00000
DATE 2017 3 20.00000
DATE 2017 3 21.00000
DATE 2017 3 22.00000
DATE 2017 3 23.00000
DATE 2017 3 24.00000
DATE 2017 3 25.00000
DATE 2017 3 26.00000
DATE 2017 3 27.00000
DATE 2017 3 28.00000
DATE 2017 3 29.00000
DATE 2017 3 30.00000
DATE 2017 3 31.00000
DATE 2017 4 1.00000
DATE 2017 4 2.00000
DATE 2017 4 3.00000
DATE 2017 4 4.00000
DATE 2017 4 5.00000
DATE 2017 4 6.00000
DATE 2017 4 7.00000
DATE 2017 4 8.00000
DATE 2017 4 9.00000
DATE 2017 4 10.00000
DATE 2017 4 11.00000
DATE 2017 4 12.00000
DATE 2017 4 13.00000
DATE 2017 4 14.00000
DATE 2017 4 15.00000
DATE 2017 4 16.00000

DATE 2017 4 17.00000
DATE 2017 4 18.00000
DATE 2017 4 19.00000
DATE 2017 4 20.00000
DATE 2017 4 21.00000
DATE 2017 4 22.00000
DATE 2017 4 23.00000
DATE 2017 4 24.00000
DATE 2017 4 25.00000
DATE 2017 4 26.00000
DATE 2017 4 27.00000
DATE 2017 4 28.00000
DATE 2017 4 29.00000
DATE 2017 4 30.00000
DATE 2017 5 1.00000
DATE 2017 5 2.00000
DATE 2017 5 3.00000
DATE 2017 5 4.00000
DATE 2017 5 5.00000
DATE 2017 5 6.00000
DATE 2017 5 7.00000
DATE 2017 5 8.00000
DATE 2017 5 9.00000
DATE 2017 5 10.00000
DATE 2017 5 11.00000
DATE 2017 5 12.00000
DATE 2017 5 13.00000
DATE 2017 5 14.00000
DATE 2017 5 15.00000
DATE 2017 5 16.00000
DATE 2017 5 17.00000
DATE 2017 5 18.00000
DATE 2017 5 19.00000
DATE 2017 5 20.00000
DATE 2017 5 21.00000
DATE 2017 5 22.00000
DATE 2017 5 23.00000
DATE 2017 5 24.00000
DATE 2017 5 25.00000
DATE 2017 5 26.00000
DATE 2017 5 27.00000
DATE 2017 5 28.00000
DATE 2017 5 29.00000
DATE 2017 5 30.00000
DATE 2017 5 31.00000
DATE 2017 6 1.00000
DATE 2017 6 2.00000
DATE 2017 6 3.00000
DATE 2017 6 4.00000

```
DATE 2017 6  5.00000
DATE 2017 6  6.00000
DATE 2017 6  7.00000
DATE 2017 6  8.00000
DATE 2017 6  9.00000
DATE 2017 6 10.00000
DATE 2017 6 11.00000
DATE 2017 6 12.00000
DATE 2017 6 13.00000
DATE 2017 6 14.00000
DATE 2017 6 15.00000
DATE 2017 6 16.00000
DATE 2017 6 17.00000
DATE 2017 6 18.00000
DATE 2017 6 19.00000
DATE 2017 6 20.00000
DATE 2017 6 21.00000
DATE 2017 6 22.00000
DATE 2017 6 23.00000
DATE 2017 6 24.00000
DATE 2017 6 25.00000
DATE 2017 6 26.00000
DATE 2017 6 27.00000
DATE 2017 6 28.00000
DATE 2017 6 29.00000
DATE 2017 6 30.00000


RESULTS SPEC 'Fracture Spacing I'
RESULTS SPEC SPECNOTCALCVAL -100000
RESULTS SPEC REGION 'Layer 1 - Whole layer'
RESULTS SPEC REGIONTYPE 'REGION_LAYER'
RESULTS SPEC LAYERNUMB 1
RESULTS SPEC PORTYPE 1
RESULTS SPEC CON 1.5
RESULTS SPEC REGION 'Sector: SRV'
RESULTS SPEC REGIONTYPE 'REGION_SECTOR'
RESULTS SPEC LAYERNUMB 0
RESULTS SPEC PORTYPE 1
RESULTS SPEC CON 0.15
RESULTS SPEC SPECKEEPMOD 'YES'
RESULTS SPEC STOP


RESULTS SPEC 'Fracture Spacing J'
RESULTS SPEC SPECNOTCALCVAL -100000
RESULTS SPEC REGION 'Layer 1 - Whole layer'
RESULTS SPEC REGIONTYPE 'REGION_LAYER'
RESULTS SPEC LAYERNUMB 1
```

```
RESULTS SPEC PORTYPE 1
RESULTS SPEC CON 1.5
RESULTS SPEC REGION 'Sector: SRV'
RESULTS SPEC REGIONTYPE 'REGION_SECTOR'
RESULTS SPEC LAYERNUMB 0
RESULTS SPEC PORTYPE 1
RESULTS SPEC CON 0.15
RESULTS SPEC SPECKEEPMOD 'YES'
RESULTS SPEC STOP

RESULTS SPEC 'Fracture Spacing K'
RESULTS SPEC SPECNOTCALCVAL -100000
RESULTS SPEC REGION 'Layer 1 - Whole layer'
RESULTS SPEC REGIONTYPE 'REGION_LAYER'
RESULTS SPEC LAYERNUMB 1
RESULTS SPEC PORTYPE 1
RESULTS SPEC CON 1.5
RESULTS SPEC REGION 'Sector: SRV'
RESULTS SPEC REGIONTYPE 'REGION_SECTOR'
RESULTS SPEC LAYERNUMB 0
RESULTS SPEC PORTYPE 1
RESULTS SPEC CON 0.15
RESULTS SPEC SPECKEEPMOD 'YES'
RESULTS SPEC STOP

RESULTS SPEC 'Permeability K' FRACTURE
RESULTS SPEC SPECNOTCALCVAL -100000
RESULTS SPEC REGION 'Layer 1 - Whole layer'
RESULTS SPEC REGIONTYPE 'REGION_LAYER'
RESULTS SPEC LAYERNUMB 1
RESULTS SPEC PORTYPE 2
RESULTS SPEC CON 0.009
RESULTS SPEC REGION 'Sector: SRV'
RESULTS SPEC REGIONTYPE 'REGION_SECTOR'
RESULTS SPEC LAYERNUMB 0
RESULTS SPEC PORTYPE 2
RESULTS SPEC CON 0.09
RESULTS SPEC SPECKEEPMOD 'YES'
RESULTS SPEC STOP

RESULTS SPEC 'Permeability I' MATRIX
RESULTS SPEC SPECNOTCALCVAL -100000
RESULTS SPEC REGION 'Layer 1 - Whole layer'
RESULTS SPEC REGIONTYPE 'REGION_LAYER'
RESULTS SPEC LAYERNUMB 1
RESULTS SPEC PORTYPE 1
```

```
RESULTS SPEC CON 9E-005
RESULTS SPEC SPECKEEPMOD 'YES'
RESULTS SPEC STOP

RESULTS SPEC 'Permeability I' FRACTURE
RESULTS SPEC SPECNOTCALCVAL -100000
RESULTS SPEC REGION 'Layer 1 - Whole layer'
RESULTS SPEC REGIONTYPE 'REGION_LAYER'
RESULTS SPEC LAYERNUMB 1
RESULTS SPEC PORTYPE 2
RESULTS SPEC CON 0.009
RESULTS SPEC REGION 'Sector: SRV'
RESULTS SPEC REGIONTYPE 'REGION_SECTOR'
RESULTS SPEC LAYERNUMB 0
RESULTS SPEC PORTYPE 2
RESULTS SPEC CON 0.09
RESULTS SPEC SPECKEEPMOD 'YES'
RESULTS SPEC STOP

RESULTS SPEC 'Porosity' MATRIX
RESULTS SPEC SPECNOTCALCVAL -99999
RESULTS SPEC REGION 'Layer 1 - Whole layer'
RESULTS SPEC REGIONTYPE 'REGION_LAYER'
RESULTS SPEC LAYERNUMB 1
RESULTS SPEC PORTYPE 1
RESULTS SPEC CON 0.11
RESULTS SPEC SPECKEEPMOD 'YES'
RESULTS SPEC STOP

RESULTS SPEC 'Porosity' FRACTURE
RESULTS SPEC SPECNOTCALCVAL -100000
RESULTS SPEC REGION 'Layer 1 - Whole layer'
RESULTS SPEC REGIONTYPE 'REGION_LAYER'
RESULTS SPEC LAYERNUMB 1
RESULTS SPEC PORTYPE 2
RESULTS SPEC CON 0.011
RESULTS SPEC REGION 'Sector: SRV'
RESULTS SPEC REGIONTYPE 'REGION_SECTOR'
RESULTS SPEC LAYERNUMB 0
RESULTS SPEC PORTYPE 2
RESULTS SPEC CON 0.015
RESULTS SPEC SPECKEEPMOD 'YES'
RESULTS SPEC STOP

RESULTS SPEC 'Permeability J' FRACTURE
```

```
RESULTS SPEC SPECNOTCALCVAL -100000
RESULTS SPEC REGION 'Layer 1 - Whole layer'
RESULTS SPEC REGIONTYPE 'REGION_LAYER'
RESULTS SPEC LAYERNUMB 1
RESULTS SPEC PORTYPE 2
RESULTS SPEC CON 0.018
RESULTS SPEC REGION 'Sector: SRV'
RESULTS SPEC REGIONTYPE 'REGION_SECTOR'
RESULTS SPEC LAYERNUMB 0
RESULTS SPEC PORTYPE 2
RESULTS SPEC CON 0.18
RESULTS SPEC SPECKEEPMOD 'YES'
RESULTS SPEC STOP

RESULTS SPEC 'Pressure' FRACTURE
RESULTS SPEC SPECNOTCALCVAL -99999
RESULTS SPEC REGION 'All Layers (Whole Grid)'
RESULTS SPEC REGIONTYPE 'REGION_WHOLEGRID'
RESULTS SPEC LAYERNUMB 0
RESULTS SPEC PORTYPE 2
RESULTS SPEC CON 3940
RESULTS SPEC SPECKEEPMOD 'YES'
RESULTS SPEC STOP

RESULTS SPEC 'Rel Perm Set Num' MATRIX
RESULTS SPEC SPECNOTCALCVAL -99999
RESULTS SPEC REGION 'All Layers (Whole Grid)'
RESULTS SPEC REGIONTYPE 'REGION_WHOLEGRID'
RESULTS SPEC LAYERNUMB 0
RESULTS SPEC PORTYPE 1
RESULTS SPEC CON 1
RESULTS SPEC SPECKEEPMOD 'YES'
RESULTS SPEC STOP

RESULTS SPEC 'Permeability J' MATRIX
RESULTS SPEC SPECNOTCALCVAL -100000
RESULTS SPEC REGION 'Layer 1 - Whole layer'
RESULTS SPEC REGIONTYPE 'REGION_LAYER'
RESULTS SPEC LAYERNUMB 1
RESULTS SPEC PORTYPE 1
RESULTS SPEC CON 9E-005
RESULTS SPEC SPECKEEPMOD 'YES'
RESULTS SPEC STOP

RESULTS SPEC 'Permeability K' MATRIX
```

```
RESULTS SPEC SPECNOTCALCVAL -100000
RESULTS SPEC REGION 'Layer 1 - Whole layer'
RESULTS SPEC REGIONTYPE 'REGION_LAYER'
RESULTS SPEC LAYERNUMB 1
RESULTS SPEC PORTYPE 1
RESULTS SPEC CON 9E-005
RESULTS SPEC SPECKEEPMOD 'YES'
RESULTS SPEC STOP

RESULTS SPEC 'Pressure' MATRIX
RESULTS SPEC SPECNOTCALCVAL -99999
RESULTS SPEC REGION 'All Layers (Whole Grid)'
RESULTS SPEC REGIONTYPE 'REGION_WHOLEGRID'
RESULTS SPEC LAYERNUMB 0
RESULTS SPEC PORTYPE 1
RESULTS SPEC CON 3940
RESULTS SPEC SPECKEEPMOD 'YES'
RESULTS SPEC STOP

RESULTS SPEC 'Water Saturation' FRACTURE
RESULTS SPEC SPECNOTCALCVAL -99999
RESULTS SPEC REGION 'Layer 1 - Whole layer'
RESULTS SPEC REGIONTYPE 'REGION_LAYER'
RESULTS SPEC LAYERNUMB 1
RESULTS SPEC PORTYPE 2
RESULTS SPEC CON 0.001
RESULTS SPEC SPECKEEPMOD 'YES'
RESULTS SPEC STOP

RESULTS SPEC 'Water Saturation' MATRIX
RESULTS SPEC SPECNOTCALCVAL -100000
RESULTS SPEC REGION 'Layer 1 - Whole layer'
RESULTS SPEC REGIONTYPE 'REGION_LAYER'
RESULTS SPEC LAYERNUMB 1
RESULTS SPEC PORTYPE 1
RESULTS SPEC CON 0.1
RESULTS SPEC SPECKEEPMOD 'YES'
RESULTS SPEC STOP

RESULTS SPEC 'Rel Perm Set Num' FRACTURE
RESULTS SPEC SPECNOTCALCVAL -99999
RESULTS SPEC REGION 'All Layers (Whole Grid)'
RESULTS SPEC REGIONTYPE 'REGION_WHOLEGRID'
RESULTS SPEC LAYERNUMB 0
RESULTS SPEC PORTYPE 2
```

```
RESULTS SPEC CON 2
RESULTS SPEC SPECKEEPMOD 'YES'
RESULTS SPEC STOP


RESULTS SPEC 'Global Composition$C' 'CH4' FRACTURE
RESULTS SPEC SPECNOTCALCVAL -99999
RESULTS SPEC REGION 'All Layers (Whole Grid)'
RESULTS SPEC REGIONTYPE 'REGION_WHOLEGRID'
RESULTS SPEC LAYERNUMB 0
RESULTS SPEC PORTYPE 2
RESULTS SPEC CON 1
RESULTS SPEC SPECKEEPMOD 'YES'
RESULTS SPEC STOP


RESULTS SPEC 'Global Composition$C' 'CH4' MATRIX
RESULTS SPEC SPECNOTCALCVAL -99999
RESULTS SPEC REGION 'All Layers (Whole Grid)'
RESULTS SPEC REGIONTYPE 'REGION_WHOLEGRID'
RESULTS SPEC LAYERNUMB 0
RESULTS SPEC PORTYPE 1
RESULTS SPEC CON 1
RESULTS SPEC SPECKEEPMOD 'YES'
RESULTS SPEC STOP


RESULTS SPEC 'Grid Top'
RESULTS SPEC SPECNOTCALCVAL -100000
RESULTS SPEC REGION 'Layer 1 - Whole layer'
RESULTS SPEC REGIONTYPE 'REGION_LAYER'
RESULTS SPEC LAYERNUMB 1
RESULTS SPEC PORTYPE 1
RESULTS SPEC CON 6500
RESULTS SPEC SPECKEEPMOD 'YES'
RESULTS SPEC STOP


RESULTS SPEC 'Grid Thickness'
RESULTS SPEC SPECNOTCALCVAL -100000
RESULTS SPEC REGION 'Layer 1 - Whole layer'
RESULTS SPEC REGIONTYPE 'REGION_LAYER'
RESULTS SPEC LAYERNUMB 1
RESULTS SPEC PORTYPE 1
RESULTS SPEC CON 200
RESULTS SPEC SPECKEEPMOD 'YES'
RESULTS SPEC STOP
```

Appendix B

CMG-Results Report RWD File Template

```
*LINES-PER-PAGE 100000

*TABLE-FOR

  *COLUMN-FOR *PARAMETERS 'Well Bottom-hole Pressure'   *WELLS 'Well-1'
*FILES 'Block 37x37_%%NUMBER%%.irf'

*TABLE-END
```

Appendix C

MATLAB Script

C1 Random Number Generation Function

```
function r = genRand(min, max, N)
% get random vector. each vector is of size of Nx1. min is the lower
limit,
% max is the upper limit, and N is the number of random number to be
% generated.
r = min + (max - min) * rand(N,1);
end
```

C2 Langmuir Adsorption Table Generation Function

```
function [output, Max]=langmuir(Vl,Pl)
pressure = linspace(14.7,6500,20)';
adsorption = 0.000598248 * Vl * pressure ./ (Pl + pressure);
data = [pressure, adsorption];
% transfer data into string
output='';
for i=1:size(data,1)
    output=[output num2str(data(i,:)) sprintf('\n')];
end
% extract the max value
Max = max(adsorption)*1.1;
end
```

C3 Write Text File Function

```
function writeTextFile(filename, content)
% open file. 'w' means 'write' mode
fileID = fopen(filename,'w');
% write content to file. '%s' means the content is text string
fprintf(fileID,'%s',content);
% close file
fclose(fileID);
end
```

C4 DAT File Generation Script

```
%% delete existing variables
clear
%% read in pattern file as string
% pattern file name
filename='Block 37x37.dat';
% read pattern file to text string
pattern = fileread(filename);
%% generate random varibles (set)
% number of random values to be generated
N = 7000;
% random variable: matrix porosity
thickness = genRand(50,200,N);
matpor = genRand(0.06,0.11,N);
frcpor = genRand(0.006,0.015,N);
srvfrcpor = genRand(0.06,0.15,N);
matperm = genRand(0.00001,0.00009,N);
frcperm = genRand(0.0003,0.009,N);
srvfrcperm = genRand(0.003, 0.09, N);
frcspac = genRand (1.5, 4, N);
srvfrcspac = genRand(0.15,0.4,N);
frc_halflength=genRand(50,200,N);
VlCH4 = genRand(50,150,N);
PlCH4 = genRand(500,1000,N);
well_length=genRand(3000,5000,N);
Pr = genRand(2000,4000,N);
q=genRand(1e6,1e7,N);

%Calculate grid block size
side_length=2*well_length;
length_a=well_length/29;
length_b=well_length*0.08333;
length_c=well_length*0.0625;
width_a=frc_halflength*2/5;
width_b=(side_length-2*frc_halflength)/4/11;
width_c=(side_length-2*frc_halflength)/4/5;

%% generate dat files
% loop for each random variabel set,
% replace the corresponding words in pattern with the generated
variable values
for i = 1 : N
    newdat = pattern;
    % replace corresponding %%variable%% in the pattern text
    newdat = strrep(newdat,'%%THICKNESS%%',num2str(thickness(i)));
    newdat = strrep(newdat,'%%MATPOR%%',num2str(matpor(i)));
    newdat = strrep(newdat,'%%FRCPOR%%',num2str(frcpor(i)));
    newdat = strrep(newdat,'%%SRVFRCPOR%%',num2str(srvfrcpor(i)));
    newdat = strrep(newdat,'%%MATPERM%%',num2str(matperm(i)));
    newdat = strrep(newdat,'%%FRCPERM%%',num2str(frcperm(i)));
    newdat = strrep(newdat,'%%SRVFRCPERM%%',num2str(srvfrcperm(i)));
    newdat = strrep(newdat,'%%FRCSPAC%%',num2str(frcspac(i)));
    newdat = strrep(newdat,'%%SRVFRCSPAC%%',num2str(srvfrcspac(i)));
```

```
    [output, adsMax] = langmuir(VlCH4(i), PlCH4(i));
    newdat = strrep(newdat,'%%ADSCH4%%',output);
    newdat = strrep(newdat,'%%ADSCH4MAX%%',num2str(adsMax));
    newdat = strrep(newdat,'%%PRESSURE%%',num2str(Pr(i)));
    newdat = strrep(newdat,'%%FLOWRATE%%',num2str(q(i)));
    newdat = strrep(newdat,'%%LENGTHA%%',num2str(length_a(i)));
    newdat = strrep(newdat,'%%LENGTHB%%',num2str(length_b(i)));
    newdat = strrep(newdat,'%%LENGTHC%%',num2str(length_c(i)));
    newdat = strrep(newdat,'%%WIDTHA%%',num2str(width_a(i)));
    newdat = strrep(newdat,'%%WIDTHB%%',num2str(width_b(i)));
    newdat = strrep(newdat,'%%WIDTHC%%',num2str(width_c(i)));

    % new file name is to attach number in the back of the pattern
filename
    newfilename = strrep(filename,'.dat',['_',num2str(i),'.dat']);
    % write new text to a new file
    writeTextFile(newfilename, newdat);
end
%% save the variables to mat file
save
```

C5 RWD File Generation Script

```
%% delete existing variables
clear
%% read in pattern file as string
% pattern file name
filename='Extract-Pattern.rwd';
% read pattern file to text string
pattern = fileread(filename);
N = 7000;

for i = 6001 : N
    newdat = pattern;
    % replace corresponding %%variable%% in the pattern text
    newdat = strrep(newdat,'%%NUMBER%%',num2str(i));
    newfilename = strrep(filename,'.rwd',['_',num2str(i),'.rwd']);
    % write new text to a new file
    writeTextFile(newfilename, newdat);
end
```

C6 Example of ANN Training Script

```
%% ANN Training
clc,clear
format long
%% Importing data
load nnTrainData.mat
%% Training
rawinput1(16,:)=[];
%Start Training
P=rawinput1;
T=rawoutput1;
%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
[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=12;
NNeu2=12;
% creating the cascade backpropagation network
net=newcf(Pn,Tn,[NNeu1,NNeu2],{'tansig','tansig','purelin'},...
    'trainscg','learngdm','msereg');
%setting training parameters for the network
net.trainParam.goal=0.0001; %accuracy within this range
net.trainParam.epochs=20000; % 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);
Tn_val_ann=sim(net,Pn_val);
%denormalizing the data sets obtained
%output reversal
T_train=mapminmax('reverse',Tn_train,ts);
T_val=mapminmax('reverse',Tn_val,ts);
```

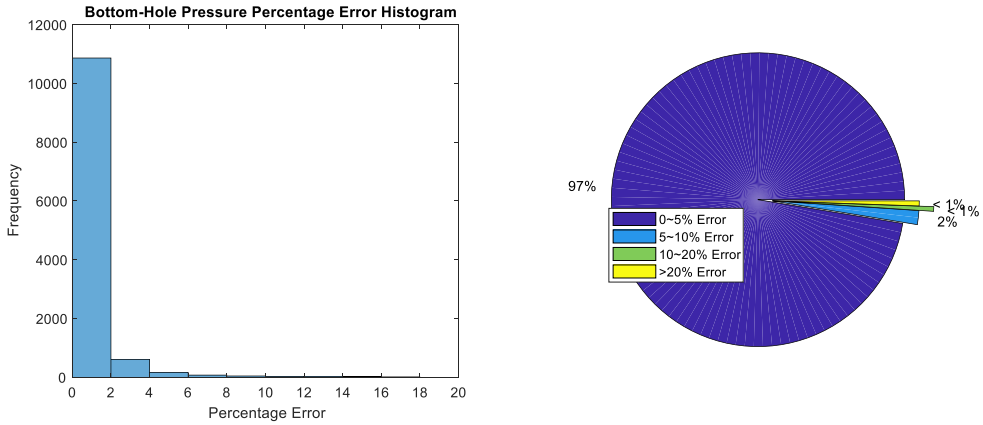
```
T_test=mapminmax('reverse',Tn_test,ts);
T_train_ann=mapminmax('reverse',Tn_train_ann,ts);
T_val_ann=mapminmax('reverse',Tn_val_ann,ts);
T_test_ann=mapminmax('reverse',Tn_test_ann,ts);
%input reversal
Pn_train=mapminmax('reverse',Pn_train,ps);
Pn_val=mapminmax('reverse',Pn_val,ps);
Pn_test=mapminmax('reverse',Pn_test,ps);
error_test=abs((T_test-T_test_ann)./T_test);
error_train=abs((T_train-T_train_ann)./T_train);
error_val = abs((T_val-T_val_ann)./T_val);
mean(mean(error_train))*100
mean(mean(error_val))*100
mean(mean(error_test))*100

save('ANN1.mat')
```

Appendix D

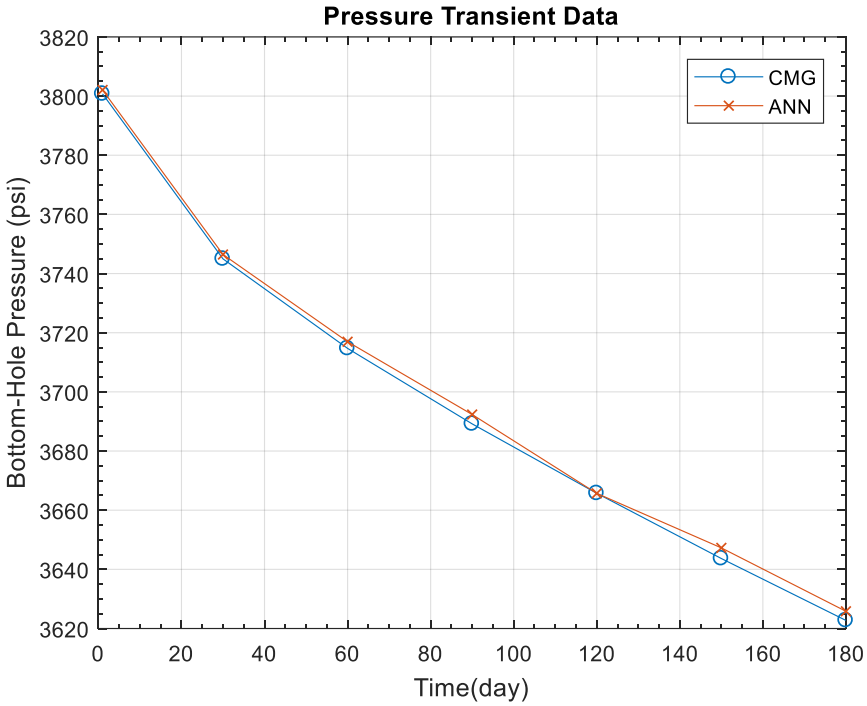
Type I ANN Results

ANN1 Average Error

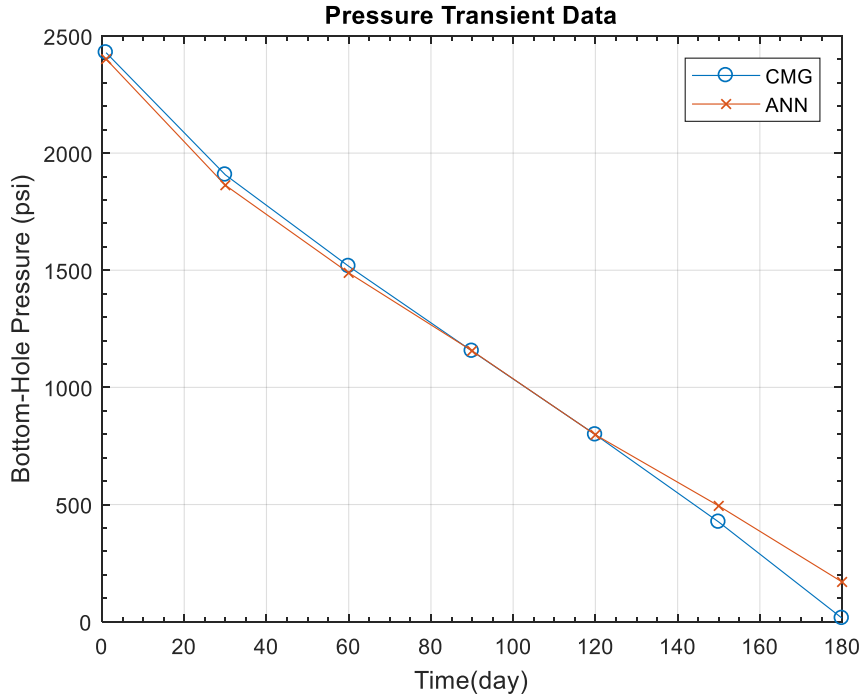
	Average Error
Training Set	1.07%
Validation Set	0.99%
Testing Set	1.30%



ANN1 Bottom-Hole Pressure Error Distribution



Best Case of ANN1 Prediction



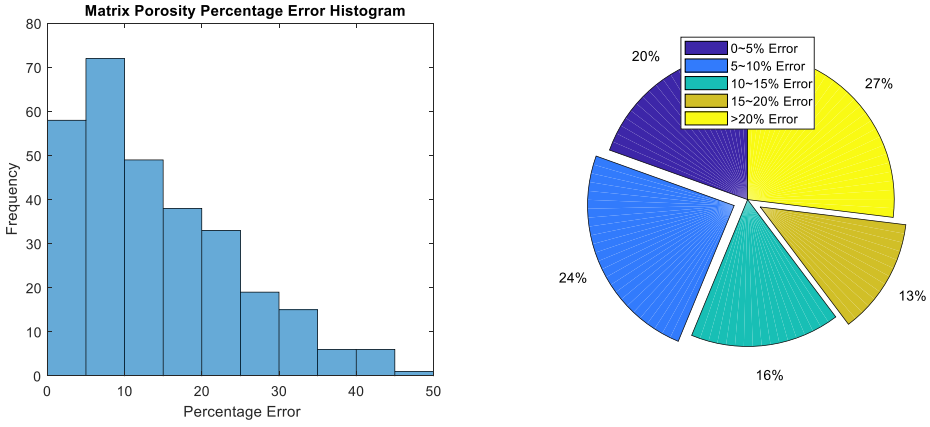
Worst Case of ANN4a Prediction

ANN3a Average Error

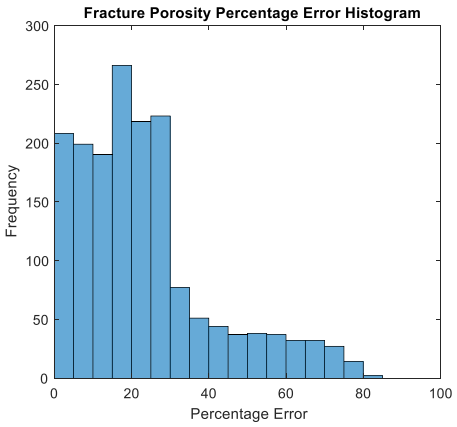
	Average Error
Training Set	22.20%
Validation Set	22.29%
Testing Set	22.77%

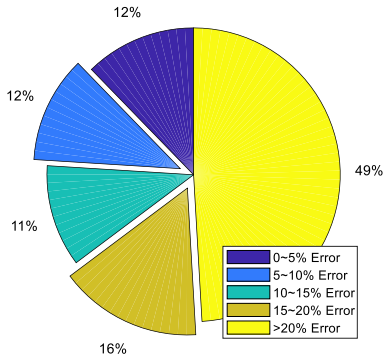
ANN3a Average Error by Parameter

	Average Error
Matrix Porosity (%)	12.96%
Fracture Porosity (%)	23.54%
SRV Fracture Porosity (%)	10.14%
Matrix Permeability (mD)	63.83%
Fracture Permeability (mD)	16.33%
SRV Fracture Permeability (mD)	4.38%
Fracture Spacing (ft)	25.40%
SRV Fracture Spacing (ft)	25.85%

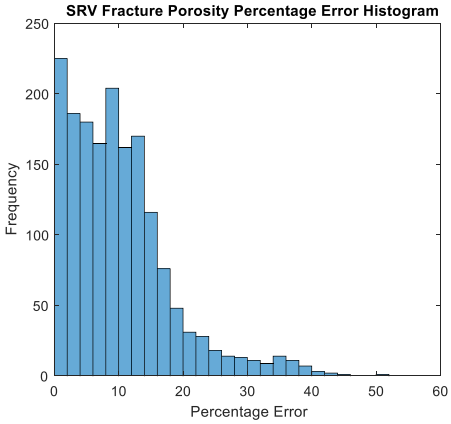


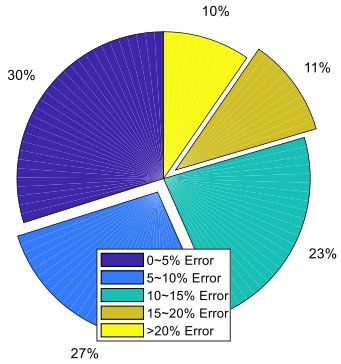
ANN3a Matrix Porosity Error Distribution



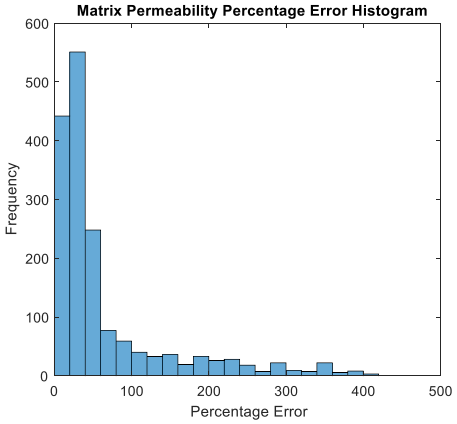


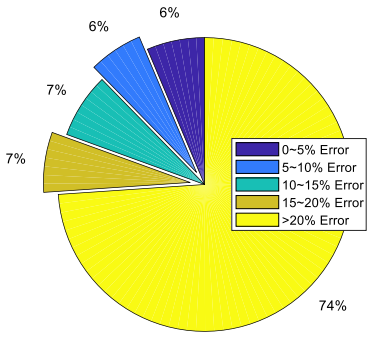
ANN3a Fracture Porosity Error Distribution



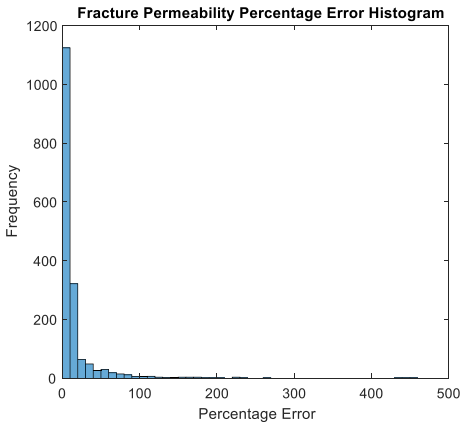


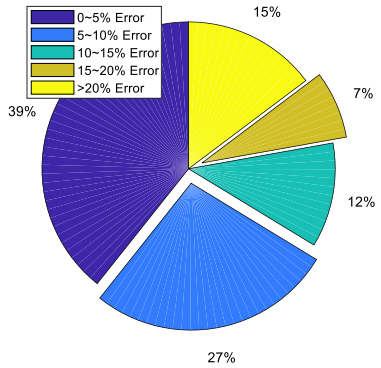
ANN3a SRV Fracture Porosity Error Distribution



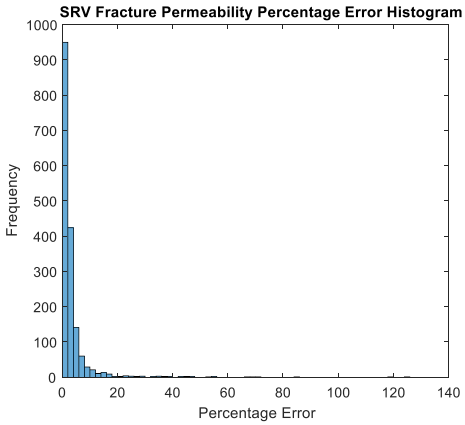


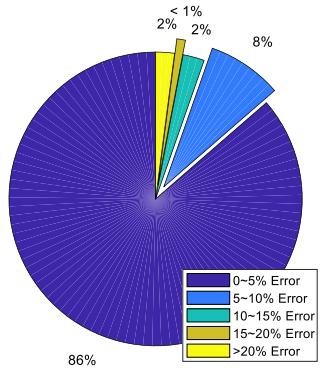
ANN3a Matrix Permeability Error Distribution



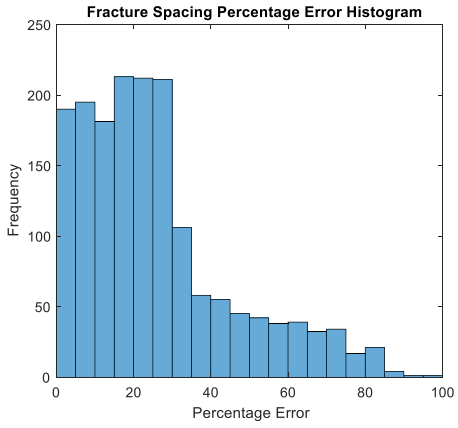


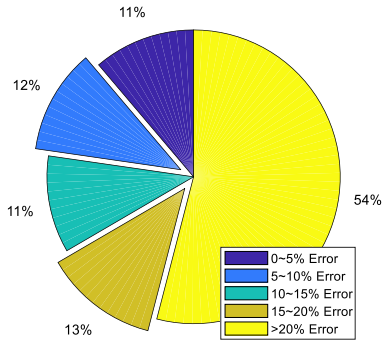
ANN3a Fracture Permeability Error Distribution



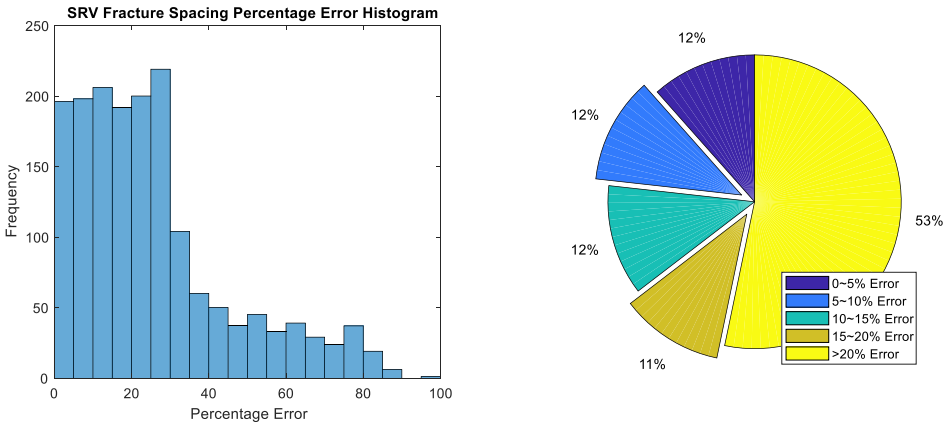


ANN3a SRV Fracture Permeability Error Distribution





ANN3a Fracture Spacing Error Distribution



ANN3a SRV Fracture Spacing Error Distribution

Best Case of ANN3a Prediction

	CMG	ANN	Average Error
Matrix Porosity (%)	0.085971	0.086525	0.64%
Fracture Porosity (%)	0.009243	0.010414	12.67%
SRV Fracture Porosity (%)	0.098785	0.098145	0.65%
Matrix Permeability (mD)	5.03E-05	4.98E-05	1.11%
Fracture Permeability (mD)	0.008101	0.008043	0.72%
SRV Fracture Permeability (mD)	0.027729	0.028009	1.01%
Fracture Spacing (ft)	2.502598	2.743549	9.63%
SRV Fracture Spacing (ft)	0.273016	0.282771	3.57%

Worst Case of ANN3a Prediction

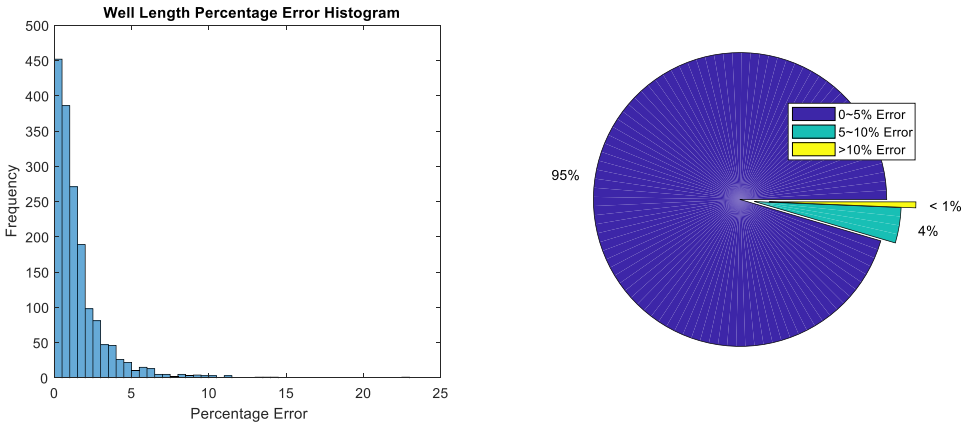
	CMG	ANN	Average Error
Matrix Porosity (%)	0.065902	0.072175	9.52%
Fracture Porosity (%)	0.006486	0.010144	56.40%
SRV Fracture Porosity (%)	0.07574	0.063077	16.72%
Matrix Permeability (mD)	2.42E-05	4.89E-05	102.10%
Fracture Permeability (mD)	0.000386	-0.00131	438.80%
SRV Fracture Permeability (mD)	0.006916	0.006044	12.60%
Fracture Spacing (ft)	3.253459	2.683032	17.53%
SRV Fracture Spacing (ft)	0.204909	0.268823	31.19%

ANN3b Average Error

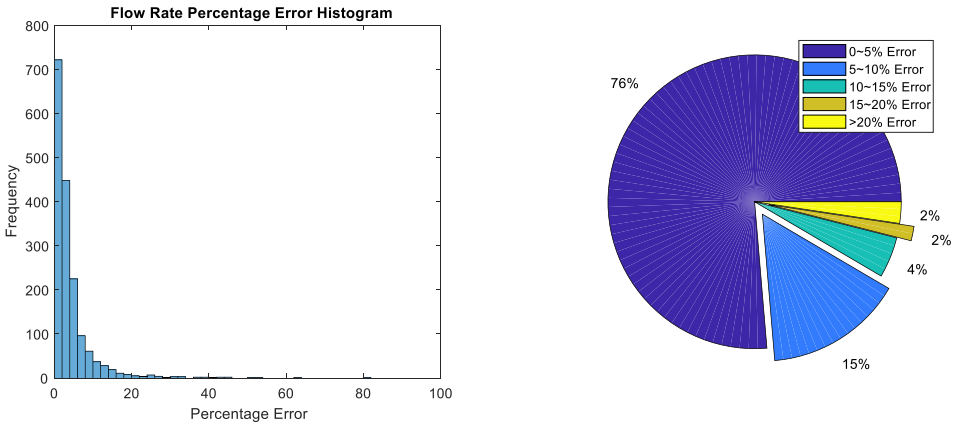
	Average Error
Training Set	3.30%
Validation Set	3.51%
Testing Set	3.27%

ANN3b Average Error by Parameter

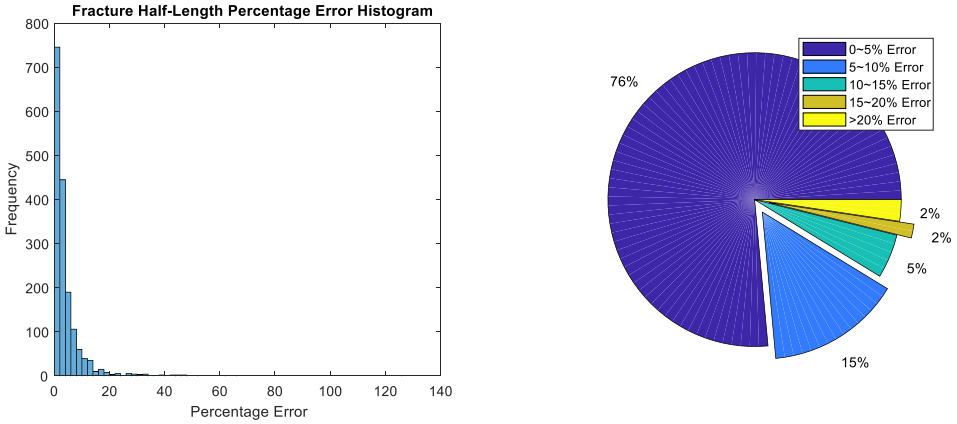
	Average Error
Well Length (ft)	2.13%
Flow Rate (mmscf/day)	3.77%
Fracture Half-Length (ft)	4.63%



ANN3b Well Length Error Distribution



ANN3b Flow Rate Error Distribution



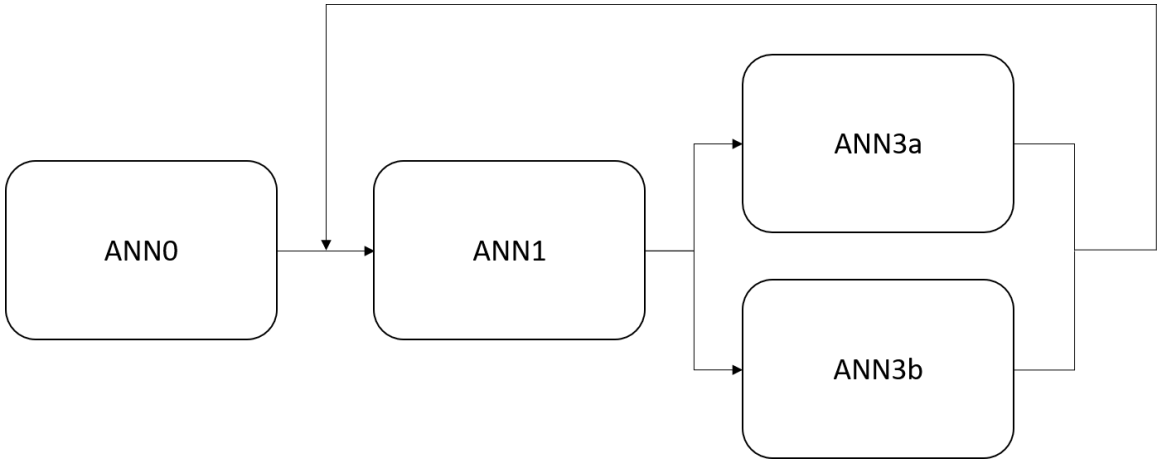
ANN3b Fracture Half-Length Error Distribution

Best Case of ANN3b Prediction

	CMG	ANN	Average Error
Well Length (ft)	4254.924	4247.228	0.18%
Flow Rate (mmscf/day)	7871663	7856931	0.19%
Fracture Half-Length (ft)	84.8154	84.81595	0.00%

Worst Case of ANN3b Prediction

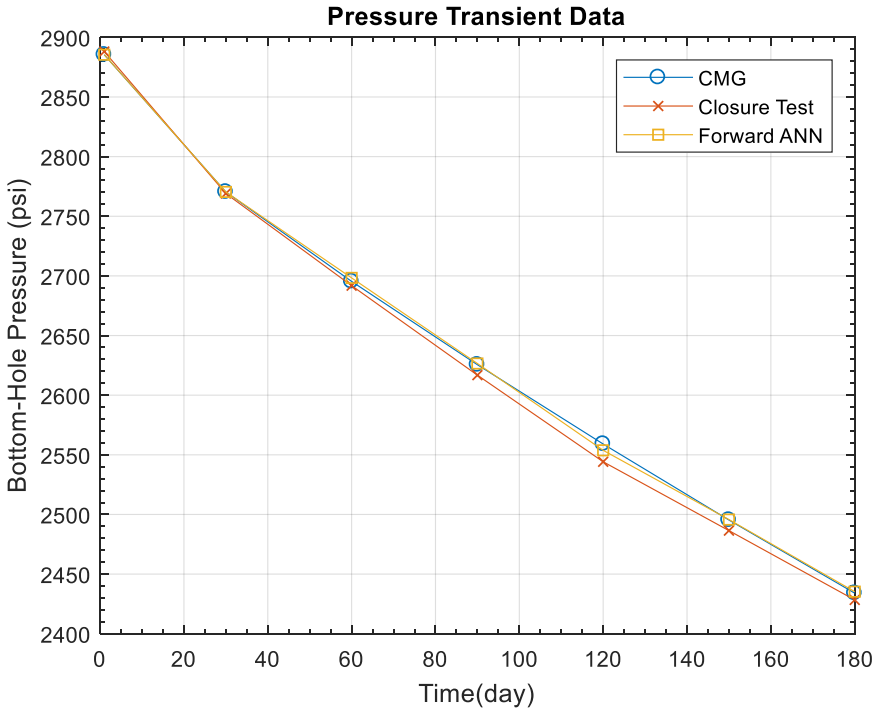
	CMG	ANN	Average Error
Well Length (ft)	3732.711	3382.183	9.39%
Flow Rate (mmscf/day)	5492593	7490277	36.37%
Fracture Half-Length (ft)	50.90593	114.2404	124.41%



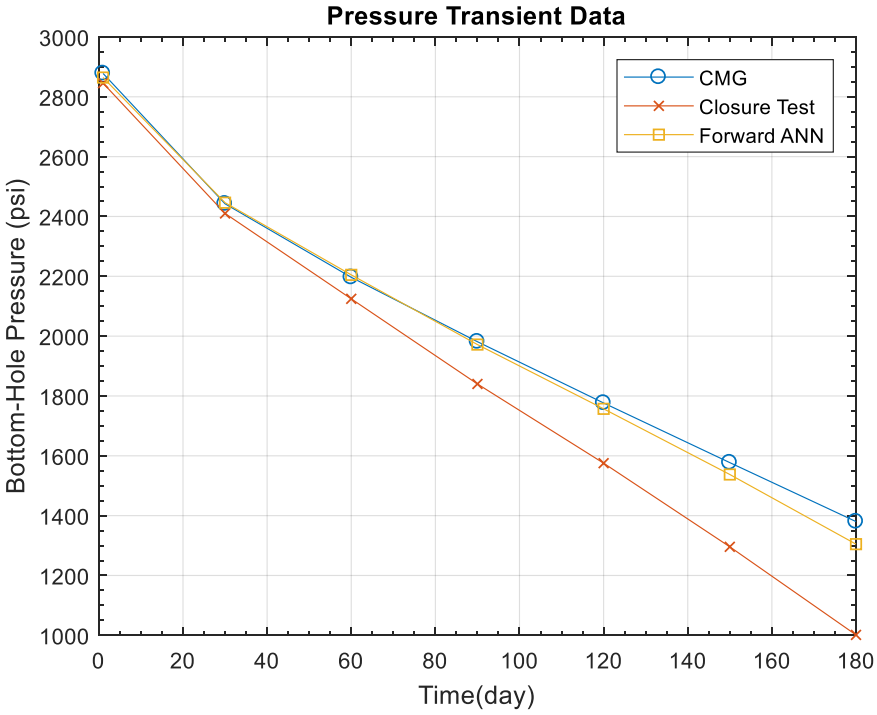
Closure Test 1 Diagram

Type I Data Closure Test Average Error of 20 Fresh Cases

Case Number	Average Error
1	2.01%
2	0.41%
3	0.64%
4	1.05%
5	3.28%
6	9.18%
7	0.48%
8	0.95%
9	3.31%
10	2.58%
11	9.92%
12	5.15%
13	0.67%
14	2.09%
15	1.34%
16	0.62%
17	1.10%
18	0.35%
19	2.75%
20	0.25%



Best Case of Type I Data Closure Test Prediction



Worst Case of Type I Data Closure Test Prediction