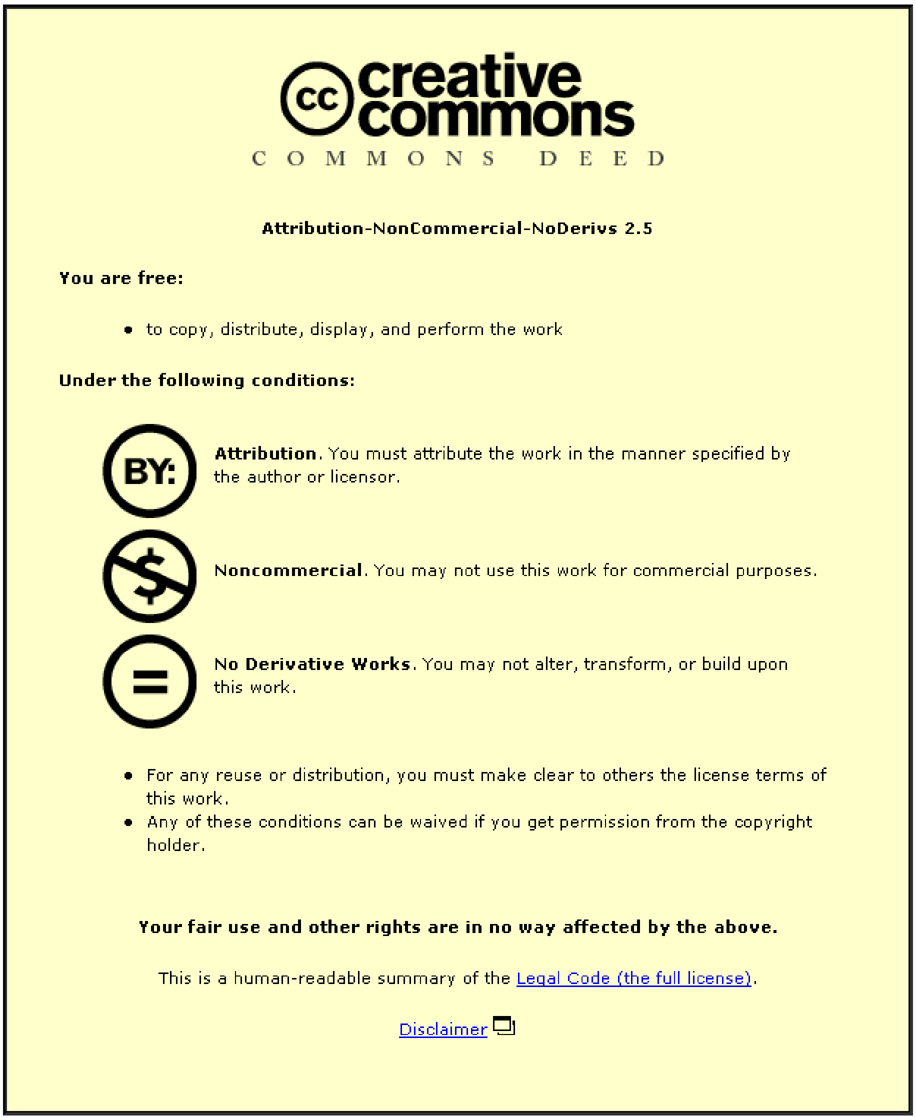


Artificial Neural Network (ANN) Modeling of Dynamic Effects on Two-phase Flow in Homogenous Porous Media

Navraj S. Hanspal

School of Mechanical, Aerospace and Civil Engineering (MACE), University of Manchester, Greater Manchester, M60 1QD, UK

&

Computational Engineering Group, Dow Corning Corporation (HSC)
North Americas, Hemlock, Michigan 48626, USA

Babatunde A. Allison

Chemical Engineering Department, Loughborough University, Loughborough LE11 3TU, Leicestershire, UK

Lipika Deka

Department of Computer Science, Indian Institute of Technology, Guwahati 110032, Assam, India

Diganta B. Das*

Chemical Engineering Department, Loughborough University, Loughborough LE11 3TU, Leicestershire, UK

Accepted for Publication in:

Journal of Hydroinformatics

September 2012

*Author for correspondence (Email: D.B.Das@lboro.ac.uk)

Artificial Neural Network (ANN) Modeling of Dynamic Effects on Two-phase Flow in Homogeneous Porous Media

Abstract

A number of recent models that describe two-phase flow processes in porous domains under the assumption of dynamic flow condition are based on the use of a dynamic coefficient (τ). The dynamic coefficients determine the speed or ease with which flow equilibrium is attained and the dependence of capillary pressure on the time derivative of saturation $(\partial S/\partial t)$. It has been shown that τ depends on a number of factors (e.g., applied boundary pressures, and media and fluid physical properties). However, varying these parameters to calculate τ poses very significant constraints on time and computational costs. Consequently, there is an increasing interest on the development of reliable approaches for determining τ which are easier to use. In addressing this issue, this paper aims to use artificial neural networks (ANN) techniques for predicting dynamic coefficient over a range of porous media and fluid physical parameters that can affect the multiphase flow behavior. The data employed for training the ANN algorithm has been acquired from computationally intensive flow physics based modeling studies undertaken previously. It is observed that the ANN modeling can appropriately characterize the relationship between the changes in the media and fluid properties, thereby ensuring a reliable prediction of the dynamic coefficient as a function of water saturation. Our results indicate that the two hidden layered ANN network may perform better in comparison to the single hidden layer ANN models for majority of the performance tests carried out in our study. However, the single hidden layered ANN model develops an inherent capability to reliably predict complex dynamic coefficient – water saturation relationships at high water saturation contents. In addition, the double hidden layered neural network models seem to perform well at low water saturation contents. In all the cases, the single and double hidden layer ANN models are better predictors in comparison to the regression models attempted in this work.

Keywords: Artificial neural network (ANN), two phase flow, porous media, dynamic coefficient, regression models

1 Introduction

Determining flow and transport behavior of non-aqueous phase liquids (NAPLs) (e.g., tetrachloroethene (PCE), polychlorinated biphenyl (PCB) and trichloroethene (TCE), creosote, soltrol) are of enormous importance in solving many subsurface contamination problems. Characterization of the flow processes involving these chemicals depends upon the flow hydrodynamics (dynamic/static), capillary/viscous forces, mobility ratios, temperature, grain size distribution, fluid properties, length scales of observation, and others. In general, modelling the two-phase flow processes requires the solution of equations for conservation of mass and momentum in conjunction with constitutive equations for capillary pressure (P^c)-saturation (S)-relative permeability (K_r) relationships. An extended version of Darcy's law is most commonly used as the governing equation of motion for the fluid phases. The conservation of mass in the two-phase system is described by an equation that ensures conservation of phase saturation which is the ratio of the volume of the fluid phase to the total pore volume in the domain. As the constitutive P^c-S relationship, models such as the Brooks-Corey (Brooks and Corey, 1964) or van Genuchten model (van Genuchten, 1980) are frequently used. Similarly, other formulation such as Brooks-Corey-Burdine formula (Brooks and Corey, 1964) exists to calculate K_r-S relationship.

The current work is limited to the study of P^c-S relationship. Physically, this relationship represents curves which are determined by taking a porous medium sample initially saturated with a wetting fluid (e.g., water) and then letting it to gradually drain off by increasing the capillary pressure at the domain boundary and displacing the wetting fluid by a non-wetting fluid (e.g., air, oil).The main theoretical definition currently used to quantify the capillary pressure is an empirical relationship obtained under equilibrium conditions between individual phase pressures in the form of

$$P_{nw} - P_w = P^c(S_w) = f(S_w) \tag{1}$$

Where P_{nw} and P_w are the average pressures of non-wetting and wetting phases respectively and S_w is the wetting phase saturation.

A number of recent studies (Das and Mirzaei, 2012; Hanspal and Das, 2012; Bottero et al., 2011; Gray and Miller, 2011; Joekar-Nisar and Hassanizadeh, 2011; O'Carroll et al., 2010; Mirzaei and Das, 2007; Das et al., 2007; Das et al., 2006; Tsakiroglou et al., 2003) that describe two-phase flow processes in porous domains under the assumption of dynamic flow condition are based on the use of a dynamic coefficient (τ). These dynamic coefficients determine the speed or ease with which flow equilibrium is attained and the dependence of capillary pressure on the time derivative of saturation $(\partial S/\partial t)$. τ establishes the speed at which flow equilibrium ($\partial S/\partial t = 0$) is reached. If τ is small, the equivalence between $P^{c,dyn}$ and $P^{c,equ}$ is established quickly. On the other hand, the necessary time period to reach the equilibrium is high for larger τ values. Thus, the dynamic coefficient (τ) behaves as a capillary damping coefficient and indicates the dynamics of the two-phase flow system. Most of the experimental and computational flow physics based techniques for determining P^c-S relationships and the corresponding dynamic effects through dynamic coefficients (τ) calculations, are very resource intensive and exceedingly time consuming for complex three-dimensional flows in homogeneous or heterogeneous porous domains (Das and Mirzaei, 2012; Hanspal and Das, 2012; Mirzaei and Das, 2007; Das et al., 2007). In order to circumvent these difficulties, we present an artificial neural network (ANN) model that can be effectively used to determine the dynamic coefficients (τ) for two-phase flow in porous media, which in this particular study is represented by PCE and water as the fluid components. The motivation to develop and apply an ANN model for two-phase flow computations results from the ability of the ANNs to impose fewer constraints on the functional form of the relationships between input and output variables, when the complexity of the systems is difficult to anticipate (Johnson and Rogers, 2000). In the following section we discuss the background of this paper in more detail.

1.1 Artificial Neural Networks (ANNs)

An artificial neural network (ANN) is a computational tool composed of simple elements operating in parallel (Demuth et al., 2008) commonly known as neurons that can simulate the working of the human brain and the nervous system learning to perform functions (an input/output map). The neurons are grouped into subsets (input, output and hidden layers) connected to one another having bias and transfer functions associated with them. Generally, networks with biases, a sigmoid layer, and a linear output layer are capable of approximating any function with a finite number of discontinuities. The weight and bias are adjustable scalar parameters of a neuron that are modified in a sequential mode, for the network to exhibit the desired behavior. The assigned weights in conjunction with the presence of hidden layers within the network help in determining complicated relationships between the input and output data.

A back-propagation algorithm is used in this work to reduce the observed error in the predicted output variables by modifying the connection weights. Standard back-propagation includes a gradient descent algorithm, like the Widrow-Hoff learning rule (Widrow, 1962) for the multiple-layer networks and nonlinear differentiable transfer functions, in which the network weights are moved along the negative of the gradient of the performance function (Khataee and Kasiri, 2010; Demuth et al., 2008). When the error measure of the network is reduced below a user-defined minimum, the training is stopped, and the connection weights are recorded and used to perform computations. There are different architectures for neural networks which consequently require different types of algorithms, but despite an apparently complex system, a neural network is relatively simple serving two important functions: *(1) pattern classifiers* and *(2) non-linear adaptive filters*. The most commonly used ANN in engineering applications is the feed-forward network (Haykin, 1999).

The presence of multiple layers of neurons with non-linear transfer functions allows the network to learn non-linear relationships between input and output vectors.

In the context of flows within porous media ANN's have been used for a variety of applications that include, e.g., prediction of gas diffusion layer properties within polymer electrolyte membrane (PEM) fuel cells (Lobato et al., 2010; Kumbur et al., 2008), prediction of dialysis performance in ultrafiltration (Godini et al., 2010), hygrothermal property characterization in porous soils (Coelho et al., 2009), oil saturation and petrophysical property predictions in oilfield sands (Boadu 2001), groundwater contamination and pollutant infiltration forecasting (Tabach et al., 2007), simulating cross-flow filtration processes (Silva and Flauzino, 2008), optimization of groundwater remediation problems (Johnson and Rogers, 2000; Rogers and Dowla, 1994), large-scale water resource management (Yan and Minsker, 2006), permeability modeling in petroleum reservoir management (Karimpouli et al., 2010), water/wastewater treatment using various homogeneous and heterogeneous nano-catalytic processes (Khataee and Kasiri, 2010), determination of stress-strain characteristics in composites (Lefik et al., 2009) and characterization of outflow parameters influencing fractured aquifers outflows (Lallahem and Mania, 2003). For example, Rogers and Dowla (1994) proposed an ANN-based groundwater management model for optimizing aquifer remediation. The flow and transport model generated a set of sample data upon which the network could be trained. The study indicated that the ANN based management solutions were consistent with those resulting from a more conventional optimization technique, which combined solute transport modeling and non-linear programming.

It is apparent from the studies presented in the literature that although most of them signify the importance and reliability of ANN techniques for implementing porous flows for a variety of engineering problems, none of them directly concerns with the need for quantifying the dynamic effects and their influence on the flow of multiple phases. Therefore, in this work for the first time, we present an ANN based framework for handling complex three-dimensional two-phase flow computations of DNAPL displacements in the presence of dynamic effects in a robust, computationally economical and a reliable fashion in comparison to sophisticated numerical methods based CFD simulators which can be enormously time consuming for large scale recurring calculations. We also discuss the development and training strategies employed for a variety of single and two hidden layer network models. Finally, the best ANN and regression model architectures are identified for reliable dynamic coefficient predictions by evaluating the simulation results and the model performance on the basis of statistical performance parameters.

2 Artificial Neural Network (ANN) Modeling and Implementation

The input or the reference data used for the ANN model development and training result from the modeling studies conducted within three-dimensional cylindrical coarse and fine sand domains (Hanspal and Das, 2012; Hanspal and Das, 2009; Das et al., 2007; Mirzaei and Das, 2007) using immiscible DNAPL displacement experiments and quasi-static/dynamic ‘water-oil’ mode simulations handled via “Subsurface Transport Over Multiple Phases” model (STOMP) (http://stomp.pnl.gov/; White and Oostrom, 2006; Nichols et al., 1997).

The input data were collected from a number of previous studies (Hanspal and Das, 2012;

Mirzaei and Das, 2007; Das et al., 2007), which indicated that these variables are important in determining the value of τ . Furthermore, expert judgments were used to choose these variables. As explained above, these data were obtained using numerical (finite volume method) simulations with the main purpose to report the significance of the dynamic effect. The data set included approximately 150 data points. The important statistics of the data sets are shown in Table 1.

Table 1

In this work, a multilayer feed-forward network trained using back-propagation training algorithm was implemented with MATLAB's ANN toolbox and used to model the complex non-linear relationship persistent amongst the dynamic coefficient and physical properties characterizing the DNAPL displacement in multiphase transport. The back-propagation algorithm used for training the feed-forward neural network problem (Demuth et al., 2008) was implemented in four sequential steps discussed in detail below:

2.1 Data Assimilation

The data described above were imported within MATLAB by using calling functions to ensure that whilst one independent variable was changed others remained constant resulting in variations in the dynamic coefficient values. This procedure was repeated to train the network on each of the independent variables so as to produce an output close enough to the target (dependent variable). The input (independent) variables are denoted by *p* whilst the output (dependent) variables are represented by the target *t*.

2.2 Network Object Creation

MATLAB's ANN toolbox was utilized to create a feed-forward network requiring three arguments before returning the network object. The network object was created after providing the input and output parameters which then initialized the weight and bias values to determine the size of the output layer. In addition, the input data was segregated into three different sets namely the, training, validation and test data in a split of 60%, 20% and 20%, respectively.

Two-layer (single hidden) and three-layer (two hidden) feed-forward networks were developed and investigated in this study. The two-layer network has the typical format of the input variables, the target and the number of hidden neurons {*p*, *t*, 3} whilst the three-layer network is characterized by two sets of hidden layer neurons {p, *t*, [3 5]}.

2.3 Network Training

After the network weights and biases were initialized, the network was trained for function approximation (non-linear regression), pattern association and pattern classification**.** During training, the weights and biases of the network were adjusted iteratively to minimize the network performance function. The default performance function for the feed-forward networks is mean square error (mse) which is the average difference of the squared errors (Demuth et al., 2008) between the network (*a*) and the target outputs (*t*). The training was carried out in MATLAB by segregating the available data into three data sets: 60% for training, 20% for validation and 20% for testing respectively. Training was conducted multiple times in conjunction with using five-fold cross-validation to ensure each of the data-point in the 150 point data-set was a part of 60% test data-set. The network training data was then utilized for recognizing the behavioural patterns in the data, validation in order to assess the network generalization and testing to provide an independent evaluation of network generalization for new data that the network had never experienced previously.

The process parameters, goal and epoch were used to determine the stopping criteria for the network training. The training was stopped when the number of iterations exceeded the epochs or if the performance function dropped below the set tolerance value. The training was carried out until there was a continued reduction in the network's error on the validation vectors. After the network memorized the training set, training was terminated. Re-initialization of the network was pursued depending upon its accuracy. In MATLAB toolbox the initial weights of nodes are assigned randomly, so repeated training may result in different ANN performance. In this work, each ANN was trained multiple times. The number of hidden neurons was varied gradually, since large neuron numbers within the hidden layer gave the network more flexibility due to multiple parameter optimization. Under-characterization was tackled by instructing the network to optimize more parameters than the number of data vectors.

Although, the training was carried out in MATLAB by segregating the available data randomly into three data sets (i.e. 60% for training, 20% for validation and 20% for testing respectively), a well-known measure (namely, s*tratified sampling*) was applied to ensure that the statistics of the testing and training data are in close vicinity.

2.4 Network Response Simulation

After the network was trained, it was re-applied to the original vectors. Network outputs were produced by incorporating the network input and the network object, and finally applied to simulate dynamic coefficient values for a range of input parameters.

3 Pre- and Post-Processing Procedures: ANN Model Training

Specific pre- and post-processing steps discussed in the subsequent section were required to train the ANN model effectively:

3.1 Pre-Processing Procedure

When the network is created using MATLAB's ANN toolbox, default processing functions are automatically assigned to the network inputs and outputs. These functions were over-ridden by adjusting the network parameters. User-defined functions were used to scale-up the network where the epoch limit was set to be 200 iterations. The {*mapstd*} function was utilized in the scale-up operation by normalizing the mean and standard deviation of the training set to be zero and one respectively. No other scaling functions or correction factors for the inputs and outputs were utilized in this work.

It was also important that we do not use too much over-fitting of the data. This was ensured as follows. First of all, we used a relatively large data set as compared to the number of points needed to plot τ-S curve. In our case, the data set was about 30-35 times greater than the typical number of points (5-6 points) one needs to plot a τ-S curve. Secondly, we used a relatively a simple ANN structure. This ensures that there is no artificial over-fitting of the data as may be observed in complex ANN structures.

3.2 Post-Processing Analysis

As an additional measure, regression analysis was carried out using the network outputs and the corresponding targets to validate the network performance.

4 ANN Model Performance Testing and Calibration

The performance of various ANN models developed in this work were analysed against standard performance parameters and criteria (Jain et al., 2001), described below. The performances were calculated using the entire dataset.

4.1 Sum squared error (SSE)

The summed square of residuals (SSE) represents the total deviation of the simulated values in comparison to the observed values. This is represented by equation (2)

$$SSE = \sum_{i=1}^{N} \left(S_{obs} - S_{cal} \right)^2 \qquad (2)$$

where N = total number of data points predicted, S_{obs} = observed value of dynamic coefficient τ and

S_{cal} = calculated value of dynamic coefficient τ.

4.2 Average absolute relative error (AARE)

The average of the relative errors (AARE) commonly expressed as a percentage were computed using equation (3). Lower values of AARE indicate better model performance.

$$AARE = \frac{1}{N}\sum_{i=1}^{N}\left|\frac{S_{cal} - S_{obs}}{S_{obs}}\right| \times 100 \tag{3}$$

4.3 Nash-Sutcliffe efficiency coefficient (E)

The Nash-Sutcliffe efficiency coefficient (E) was computed using equation (4). Values of E nearing 1.0 depict perfect match between the observed data and outputs, signifying high model accuracy

$$E = 1 - \frac{\sum\left(S_{cal} - S_{obs}\right)^2}{\sum\left(S_{obs} - \bar{S}_{obs}\right)^2} \tag{4}$$

where $\bar{S}_{obs}$ = average observed dynamic coefficient τ

4.4 Pearson product moment coefficient of correlation (R)

Pearson product moment coefficient (R), computed using equation (5), was used to characterize the strength of linear dependency in the relationship between simulated and observed data. Values of R nearing unity indicate a good model.

$$R = \frac{\sum\left(S_{obs} - \bar{S}_{obs}\right) \times \left(S_{cal} - \bar{S}_{cal}\right)}{\sqrt{\sum\left(S_{obs} - S_{obs}\right)^2 \sum\left(S_{cal} - \bar{S}_{cal}\right)^2}} \tag{5}$$

where $\bar{S}_{cal}$ = average calculated dynamic coefficient (τ)

4.5 Threshold statistics (TS)

The threshold statistics for a level of absolute relative error (ARE) of $x\%$ were computed using equation (6) to quantify the consistency in the prediction errors (Jain and Ormsbee, 2002). Large values of threshold statistics indicate better model performance.

$$TS = \frac{N_x}{N} \tag{6}$$

where N_x = number of data points predicted for which the average relative error (ARE) is less than $x\%$.

5 Regression Modeling of Dynamic Coefficient

Linear and non-linear regression models were also developed as a part of this study to make comparisons against the predictions obtained using the ANN model. MATLAB was utilized for all the regression modeling work carried out, which is described below. The regressions were calculated using the entire dataset.

5.1 Linear multiple regression

The dynamic coefficient τ was regressed against the independent variables i.e. water saturation, viscosity ratio, density ratio, permeability and temperature using equation (7).

$$\tau = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \beta_3 x_3 + \beta_4 x_4 + \beta_5 x_5 \tag{7}$$

where τ is the dynamic coefficient, $\beta_0 \to \beta_5$ are the regression coefficients to be estimated, and $x_1 \to x_5$ are the independent variables.

Since the resulting system of equations was over-determined (William, 2005). *Left division method* based on Gauss elimination and least square techniques were used for determining the matrix coefficients which best fit the data-sets. Using this technique, the data is arranged in a matrix represented using equation (8)

$$X\beta = \tau \tag{8}$$

where

$$\beta = \begin{bmatrix} \beta_0 \\ \beta_1 \\ \cdots \\ \beta_5 \end{bmatrix} \qquad X = \begin{bmatrix} 1 & x_{11} & x_{21} & x_{31} & x_{41} & x_{51} \\ 1 & x_{12} & x_{22} & x_{32} & x_{42} & x_{52} \\ \cdots & \cdots & \cdots & \cdots & \cdots\cdot & \cdots \\ 1 & x_{1N} & x_{2N} & x_{3N} & x_{4N} & x_{5N} \end{bmatrix} \qquad \tau = \begin{bmatrix} \tau_0 \\ \tau_1 \\ \cdots \\ \tau_N \end{bmatrix}$$

$x_{1i} \to x_{5i}$ and τ_i represent the data, *i = 1, 2, 3*…….*N* where *N* is the number of data points. The solution for the coefficients can be computed using equation (9)

$$\beta = X \setminus \tau \tag{9}$$

where \ represents a backslash operator which performs matrix left division. *x* minimizes $norm\left(X^*\beta - \tau\right)$ the length of the vector $X\beta - \tau$ (Demuth et al., 2008).

5.2 Non-Linear Multiple Regression

Using similar variables as in case of linear regression, polynomials of various orders (Jain and Indurthy, 2003) represented by equations (10)-(12) were used to regress the dynamic coefficient against water saturation, viscosity ratio, density ratio, permeability and temperature.

$$\tau = \beta_0 + \beta_1(x_1)^2 + \beta_2(x_2)^2 + \beta_3(x_3)^2 + \beta_4(x_4)^2 + \beta_5(x_5)^2 \quad (10)$$

$$\tau = \beta_0 + \beta_1(x_1)^{0.05} + \beta_2(x_2)^{0.05} + \beta_3(x_3)^{0.05} + \beta_4(x_4)^{0.05} + \beta_5(x_5)^{0.05} \quad (11)$$

$$\tau = \beta_0 + \beta_1(x_1)^4 + \beta_2(x_2)^4 + \beta_3(x_3)^4 + \beta_4(x_4)^4 + \beta_5(x_5)^4 \quad (12)$$

Function based on *Gauss-Newton* algorithm with *Levenberg-Marquardt* modifications for global convergence (Demuth et al., 2008) were used to determine the *least-squares parameter* estimates.

6 Results and Discussions

The reference data used in developing and training various neural network models for predictive modeling of dynamic coefficients through incorporation of dynamic effects can be referred to in the works of Hanspal and Das (2012), Das et al. (2007) and Mirzaei and Das (2007). As described in the *data assimilation* section the data comprises of five independent and one dependent output parameter

6.1 Artificial Neural Network (ANN) Models

Two different types of ANN models were developed in this work, which include: *(1) single-hidden layer model (2) two-hidden layer model*. For each type of model the number of neurons in the input layer and the output layer were kept to be the same. The number of neurons in the hidden layer was determined using a trial-by-error procedure proposed by Jain and Indurthy (2003). The optimal model architecture was determined by varying the number of hidden neurons from 3 to 17 and performing a post-training analysis on each of the network models. As discussed before, the slope (m), correlation coefficient (r) and intercept (c) values in the proximity of unity and zero respectively, indicates an optimal model. The model training and post-training regression analysis plots for the best ANN networks developed in this work can be seen in Figures 1-4.

Figure 1(a) **Figure 1(b)**

Figure 2(a) **Figure 2(b)**

Figure 3(a) **Figure 3(b)**

Figure 4(a) **Figure 4(b)**

The mean squared errors presented in Figures 1(a)-4(a) gradually decrease as the learning and the training process continues. The performance goal indicates that the convergence of the mean squared error training, validation and test plots was set to the default mode of zero. The number of iterations (epochs) were different in all the distinct models reported, since the validation test stops the network training, when the peak performance is attained. Figures 1(b)-4(b) represent the post-training regression analyses of the network models depicting the perfect line, outputs = targets (*Y*=*T*) and the best linear regression line for the data points. The best linear regression line is then used for evaluating the slope, correlation coefficient and the intercept. Table 2 presents the performance of the ANN models on the basis of post-training line regression plots used in the determination of the number of hidden neurons which produce the most accurate fit.

Table 2

From the slopes and the correlation coefficients presented in Table 2, it is clearly observed that the network is best trained when there are 7 or 9 hidden neurons for the single hidden layer model structure and a combination of [9 11], [13 15] or [11 13] hidden neurons for the two hidden layer models. Only these models have their correlation coefficients and slopes above 0.9. In terms of the correlation coefficient, the two hidden layer structure ANN [11 13] performed best with a value of 0.9632 whilst the ANN [13 15] model structure performed best with respect to the slope with a value of 0.9595. In comparison to the two-layer hidden networks the single-layer networks ANN [7] and ANN [9] models have correlation coefficient values of 0.9578 and 0.9439, respectively.

6.2 Regression Models

The values of the regression coefficients for the linear and non-linear regression models used in this work are presented in Table 3.

Table 3

The performances of the regression models were further evaluated and compared to the ANN model performances, enlisted in the subsequent section using the criteria described in sections (4.1)-(4.5).

6.3 Model performance Criteria Evaluation

Model performance parameters (AARE, SSE, R, E and TS) were computed to determine the performance of the ANN and regression models.

Plots presented in Figures (5)-(8) provide a better means to evaluate and compare the performance of the ANN and regression models.

Figure 5	**Figure 6**
Figure 7	**Figure 8**

From the *absolute average relative error (AARE)* tests, characterized using Figure 5 it can be depicted that the regression models performed badly with the linear regression model performing the worst. The best of the single-layer and two hidden layer structures performed almost identically.

Figure 6 illustrates the comparison of the *sum squared errors (SSE)* signifying that the regression models perform poorly with the non-linear regression model-2 performing the worst (7.13333). The ANN [7] and ANN [11 13] performed best with the latter having a better value of (0.7175).

Comparisons using efficiency (E) and correlation coefficient (R) presented in Figure 7 re-illustrates the underperformance of regression models with the non-linear regression model-2 performing the worst. The ANN [7] and ANN [11 13] models have the top-most performance with the efficiency (E) and correlation coefficient (R) for ANN [7] being 0.9171 and 0.9578. ANN [11 13] performed slightly better with values of 0.9272 for E and 0.9632 for R, respectively.

From the *Threshold Statistics (TS)* plot comparison using Figure 8 it is adjudged that the regression models again perform poorly with the non-linear regression model-2 having the lowest value for threshold statistic (TS-5). The ANN [7] model on the other hand performed best for TS-100 with a value of 88.64 while the ANN [9 11] model performed best for TS-5 with a value of 65.91.

In general, it is concluded that the regression models generally perform deficiently in comparison to ANN models. The ANN [7] in the category of single hidden layer models and the ANN [11 13] within the class of two hidden layer models have the best performance levels. Two hidden layer network models performed slightly better in comparison to the single layer network models as they have better performance comparison in all the tests, except the threshold statistics, where they had lower values in comparison to the single layer network illustrated in Table 4.

Table 4

6.4 Model Simulations: Dynamic Coefficient-Water Saturation Relationship

Typically the functional relationships between the dynamic coefficient and water saturation can be characterized by smooth curves which exemplify decreasing dynamic coefficients values for increasing water saturation. Dynamic coefficients values were obtained using the best ANN and regression models developed in this work and compared against the corresponding target values obtained using the reference data (Hanspal and Das 2012, Hanspal and Das 2009, Das et al. 2007, Das et al. 2006, Mirzaei and Das 2007). Dynamic coefficient versus water saturation plots presented in Figures (9)-(11) were developed using the simulated data resulting from single, double hidden layered ANN and regression model simulations. Comparisons have been made for determining the forecasting accuracy of the ANN and regression models with regards to a typical inverse relation between the dynamic coefficient and water saturation.

Figures 9(a) and 9(b) demonstrate the predictive capabilities of the linear and non-linear regression models which are deemed to be poor. The non-linear model performs better in comparison to the linear regression model but still fails to accurately represent the characteristic behavior of the reference data.

Figure 9(a) **Figure 9(b)**

Single layer ANN model predictions compare very well against the reference data illustrated in Figures 10(a) and 10(b) respectively.

Figure 9(a) **Figure 9(b)**

The network model with 7 hidden neurons, successfully simulates the dynamic coefficient values which fall in close proximity of the reference data with fewer errors in comparison to ANN model with 9 neurons in the hidden layer. The simulations carried our using the two hidden layer ANN models are presented in Figures 11(a) and 11(b), respectively.

Figure 11(a) **Figure 11(b)**

The simulated data represents predictions of the dynamic coefficient values and characteristics of the inverse dynamic coefficient-water saturation relationship, which again compare well against the reference data. However, there are more wayward data points in comparison to the single hidden layer ANN model predictions. Simulations resulting from ANN [11 13] hidden neuron model structure compare much better in comparison to the ANN [9 11] model. Finally, the best performing single hidden layer ANN [7] and two hidden layer ANN [11 13] models from all the simulations and performance analysis are compared in Figure 12.

Figure 12

Observing Figure 12, it can be concluded that the double layer hidden ANN [11 13] model contains more prediction errors in comparison to the single hidden layer ANN [7] model. Closer inspection reveals that the two hidden layer ANN network performs better for low water saturation values enabling prediction of high dynamic coefficient values which closely resemble the reference data determined by immiscible displacement experiments and complex three-dimensional flow physics based CFD computations (Hanspal and Das, 2012). As the water saturation starts increasing the single hidden layer ANN [7] model better predicts the dynamic coefficient variations in comparison to the double hidden layer ANN [11 13] model. The ANN [11 13] model had a lot of wayward values and this was the same for all the two hidden layer models. The network structure of the best performing models include: a) single layer ANN model containing 5 input, 7 hidden layer and 1 output neuron b) double layer ANN model containing 5 input, [11 13] hidden layer and 1 output neuron which can reliably be used for predicting Dynamic Coefficient-Water Saturation relationships are presented in Figures 13 and 14.

Figure 13

Figure 14

7 Conclusions

In this work we demonstrate the successful application of ANN (single and multiple hidden layered) and regression modeling techniques (linear and non-linear) for determining complex relationships between the dynamic coefficient and the physical parameters characterizing the porous medium and the fluid properties. The data deployed for model development, network training, performance evaluation and subsequent analysis was acquired from computational physics based modeling studies (Hanspal and Das, 2012; Hanspal and Das, 2009; Mirzaei and Das, 2007). It is demonstrated that, significant computational savings can be attained using the ANN models (Figures 13-14) in comparison to the flow physics based CFD simulators. These cost savings are reflective of the reduced simulation time-scales required for determining the complex functional relationships for dynamic coefficient variations resulting from dynamic effects within multiphase flows. It was also ensured that there is not excessive over fitting of the data.

It is concluded that ANNs can model the behavioral relationship between the changes in the media and fluid properties, reliably predicting dynamic coefficients in comparison to regression models. From the performance statistics parameters which comprise of the average absolute relative error, sum squared errors and the efficiency, the two hidden layer ANN model seem to perform better in comparison to single hidden layer ANN model with similar threshold statistics in both the cases. However, from the simulation plots it was determined that single hidden layer ANN [7] model is a better predictor for high water saturation content in comparison to the double hidden layer ANN [11 13] model whilst at low water saturation ANN [11 13] performs more reliably. In most cases, however, the differences in the model predictions were small. Therefore, one could safely conclude that a well trained and validated single layered ANN structure should suffice for most practical work.

Results from this work are conclusive of the fact that ANN models operating within a hybrid framework of both single and double hidden layered neurons for different range of water saturation contents will further boost the methods for parameters estimation and the simulation characteristics.

8 Acknowledgements

We acknowledge the financial support from the EPSRC, UK for funding the project GR/S94315/01, *Micro-Heterogeneity Effects on Dynamic Capillary Pressure–Saturation Relationships in Porous*

Media, which enabled us to conduct the CFD modeling studies to generate the reference data used for the ANN simulations in this work. One of the authors (Diganta Bhusan Das) is very grateful to Prof Ashu Jain (IIT Kharagpur) for many helpful communications on the topic of the paper.

9 References

Abriola LM, Pinder GF. A multiphase approach to the modeling of porous media contamination by organic compounds. 2. Numerical simulation. Water Resources Research. 1985; 21: 19-26.

Boadu FK. Predicting oil saturation from velocities using petrophysical models and artificial neural networks. Journal of Petroleum Science and Engineering. 2001; 30:143-154.

Bottero, S, Hassanizadeh, SM, Kleingeld, PJ, Heimovaara, TJ, (2011). Non-equilibrium capillarity effects in two-phase flow through porous media at different scales. Water Resources Research, Volume: 47, Article Number: W10505, DOI: 10.1029/2011WR010887

Brooks, R. H., and A. T. Corey, Hydraulic properties of porous media, *Hydrol. Pap. 3*, Colo. State Univ., Fort Collins, 1964.

Chen CH. Committee-machine-based models for permeability prediction. Doctoral Dissertation, National Cheng Kung University of Taiwan, 2006.

Coelho LS, Freire RZ, Santos GH, Mendes N. Identification of temperature and moisture content fields using a combined neural network and clustering method approach. International Communications in Heat and Mass Transfer, 2009; 36: 304–313.

Das, DB, Mirzaei, M (2012). Dynamic effects in capillary pressure relationships for two-phase flow in porous media: Experiments and Numerical Analyses. AIChE Journal, DOI: 10.1002/aic.13777 (in press).

Das DB, Gauldie R, Mirzaei M. Dynamic effects in capillary pressure relationships for two-phase flow in porous media: Implications of fluid properties. AIChE Journal. 2007; 53(10):2505-2520.

Das DB, Mirzaei M, Widdows N. Non-uniqueness in capillary pressure-saturation-relative permeability relationships for two-phase flow in porous media: Interplay between distribution and intensity of micro-scale heterogeneities. Chemical Engineering Science. 2006; 61: 6786 – 6803.

Demuth H, Beale M, Hogan M. Neural Network Toolbox 6: User's Guide, Math Works, Inc., Version 6.0.1, 2008; 1-24.

Godini HR, Ghadrdan M, Omidkhah MR, Madaeni SS. Part II: Prediction of the dialysis process performance using Artificial Neural Network (ANN). Desalination. 2011; 265: 1-3

Gray, WG, Miller, Cass T (2011): TCAT analysis of capillary pressure in non-equilibrium, two-fluid phase, porous medium systems. Advances in Water Resources, 34(6), 770-778, **DOI:** 565 10.1016/j.advwatres.2011.04.001

Hanspal N, Das DB. Dynamic Effects on Capillary Pressure-Saturation Relationships for Two-Phase Porous Flow: Implications of Temperature. AIChE Journal, 2012, DOI: DOI 10.1002/aic.12702

Hanspal N, Das DB. Dynamic Effects for Two-Phase Flow in Porous Media: Temperature Effects, AIChE Summer meeting Tampa USA, 2009

Haykin S. Neural Networks - A Comprehensive Foundation, 2nd ed. Englewood Cliffs, NJ, Prentice-Hall, 1999.

Jain A, Indurthy P. Comparative Analysis of Event-based Rainfall-runoff Modelling Techniques-Deterministic, Statistical and Artificial Neural Networks. Journal of Hydrological Engineering. 2003; 10.1061: 1084-0699.

Jain A, Ormsbee LE. Evaluation of short-term water demand forecast modelling techniques: Conventional v/s artificial intelligence. J. Am. Water works Assoc. 2002; 94 (7):64-72.

Jain A, Varshney AK, Joshi UC. Short-term water demand forecast modelling at IIT Kanpur using Artificial Neural Networks, Water resources management. 2001; 15(5):299-321.

Johnson VM, Rogers LL. Accuracy of neural network approximators in simulation-optimization. Journal of Water Resources Planning and Management (ASCE). 2000; 126:48-56.

Joekar-Nisar, V., Hassanizadeh, SM (2011). Effect of fluids properties on non-equilibrium capillarity effects: Dynamic pore-network modelling, International Journal of Multiphase Flow, 37, 198–214

Karimpouli S, Fathianpour N, Roohi J. A new approach to improve neural networks' algorithm in permeability prediction of petroleum reservoirs using supervised committee machine neural network (SCMNN). Journal of Petroleum Science and Engineering. 2010; 73: 227–232

Khataee AR, Kasiri MB. Artificial neural networks modeling of contaminated water treatment processes by homogeneous and heterogeneous nanocatalysis. Journal of Molecular Catalysis A:Chemical. 2010; 331: 86–100

Kueper BH, Frind, EO. Two-phase flow in heterogeneous porous media. 2. Model Application. Water Resources Research. 1991; 27: 1059-1070.

Kumbur EC, Sharp KV, Mench MM. A design tool for predicting the capillary transport characteristics of fuel cell diffusion media using an artificial neural network. J Power Sources 2008;176:191-9.

Lallahem S, Mania J. A Nonlinear Rainfall-Runoff Model using Neural Network Technique: Example in Fractured Porous Media. Mathematical and Computer Modelling. 2003; 37: 1047-1061

Lefik M, Boso DP, Schrefler BA. Artificial Neural Networks in numerical modelling of composites. Comput. Methods Appl. Mech. Engrg. 2009; 198: 1785–1804

Lobato J, Canizares P, Rodrigo MA, Piuleac CG, Curteanu S, Linaresa JJ. Direct and inverse neural networks modelling applied to study the influence of the gas diffusion layer properties on PBI-based PEM fuel cells. Intl. J. of Hyd Energy. 2010; 35: 7889-7897

Mirzaei M, Das DB. Dynamic effects in capillary pressure saturations relationships for two-phase flow in 3D porous media: Implications of micro-heterogeneities. Chemical Engineering Science. 2007; 62:1927-1947.

Nichols WE, Aimo NJ, Oostrom M, White MD. STOMP application guide. PNNL-11216, UC-2010, 1997.

O'Carroll, DM, Mumford, KG, Abriola, LM, Gerhard, JI. Influence of wettability variations on dynamic effects in capillary pressure, Water Resources Research, 2010; 46: 578 10.1029/2009WR008712

Peterson C, Soderberg B. Artificial neural networks. In: Aarts, E., Lenstra, J.K. (Eds.), Local Search in Combinatorial Optimization. John Wiley & Sons, New York, 1997; 173-213.

Rogers LL, Dowla FU. Optimization of groundwater remediation using artificial neural networks with parallel solute transport modeling. Water Resources Research 30. 1994; 457-481.

Silva IN, Flauzino RA. An approach based on neural networks for estimation and generalization of crossflow filtration processes. Applied Soft Computing. 2008; 8: 590–598.

Tabach EL, Lancelot L, Shahrour I, Najjar Y. Use of artificial neural network simulation metamodelling to assess groundwater contamination in road project. Mathematical and Computer Modelling. 2007; 45: 766–776.

Tsakiroglou, CD, Theodoropoulou, MA, Karoutsos, V (2003). Nonequilibrium capillary pressure and relative permeability curves of porous media. AIChE Journal, 49(10), 2472-2486, **DOI:** 618 10.1002/aic.690491004.

White MD, Oostrom M. STOMP 4.0 User Guide. PNNL-17782, 2006.

Widrow B. Generalization and Information Storage in Networks of Adaline Neurons. In M.C. Jovitz, G.T. Jacobi & G. Goldstein (Eds.), Self-Organizing Systems, Washington DC: Spartan Books. 1962; 435–461.

William JP III. Introduction to MATLAB for Engineers (Chapter 5, Advanced Plotting and Model Building), McGraw-Hill, Singapore, International Edition. 2005; 259-334.

van Genuchten, M. T., A closed-form equation for predicting the hydraulic conductivity of unsaturated soils, *Soil Sci. Soc. Am. J.*, *44*, 892–898, 1980.

Yan SQ, Minsker B. Optimal groundwater remediation design using an adaptive neural network genetic algorithm. Water Resources Research. 2006; 42: W05407.

Table 1. Important statistics of the variables used in this study

	Independent Variable 1: Water Saturation (S_w)	Independent Variable 2: Viscosity Ratio $(V_r = \mu_{nw}/ \mu_w)$	Independent Variable 3: Density Ratio $(D_r = \rho_{nw}/\rho_w)$	Independent Variable 4: Permeability (K, m^{2-})	Independent Variable 5: Temperature (T, ^{0}C)	Dependent Variable: Dynamic Coefficient (τ) Pa.s
Range	0.105 – 0.464	0.5 - 2	0.5 - 2	5.00E-11 - 5.00E-09	20 - 80	2.82E+5 – 1.05E+11
Arithmetic Average value	0.257	0.946	1.39	3.43E-09	24.55	1.26E+10
Standard Deviation	0.091	0.32	0.47	2.31E-09	13.89	2.69E+10

Table 2: ANN model performance determined using post-training regression analysis

Number of Hidden Layers	**Artificial Neural Network Model**	**Correlation Coefficient (r)**	**Slope (m)**
1	ANN (5-3-1)	0.8396	0.7363
1	ANN (5-5-1)	0.8950	0.8359
1	ANN (5-7-1)	0.9578	0.9183
1	ANN (5-9-1)	0.9439	0.9056
1	ANN (5-11-1)	0.8634	0.7134
1	ANN (5-13-1)	0.8752	0.7715
2	ANN (5-3-5-1)	0.9455	0.8705
2	ANN (5-5-7-1)	0.9499	0.8786
2	ANN (5-7-9-1)	0.9492	0.8722
2	ANN (5-9-11-1)	0.9489	0.9025
2	ANN (5-11-13-1)	0.9632	0.9077
2	ANN (5-13-15-1)	0.9243	0.9595
2	ANN (5-15-17-1)	0.9248	0.8938

Table 3: Linear and non-Linear regression coefficients

Regression parameter	Independent variable	Linear Multiple Regression (LR) Model	Non-linear Multiple Regression (NLR) Models			
			Model 1 (NLR 1)	Model 2 (NLR 2)	Model 3 (NLR 3)	Model 4 (NLR 4)
β_0	Constant	0.5223	9.1625	0.1785	-0.6614	6.9875
β_1	x_1	-0.8281	-9.5799	-0.6248	1.4945	-4.1225
β_2	x_2	-0.0983	-1.4919	0.0051	0.2166	-11.7713
β_3	x_3	0.0791	0.5146	0.0930	-0.1092	0.8174
β_4	x_4	-0.0009	-0.0248	0.0060	0.0044	-2.0980
β_5	x_5	0.1610	1.2602	0.1471	-0.22358	1.1004

Table 4: Model performance comparisons for ANN [7] and ANN [11 13], the best performing single and double hidden layered ANN structures

MODEL	R	E	SSE	AARE	TS-5	TS-10	TS-25	TS-50	TS-100
ANN-7	0.9578	0.9171	0.7175	31.2250	57.58	63.65	76.52	84.85	88.64
ANN-[11 13]	0.9632	0.9272	0.6306	27.4732	59.85	65.91	75.00	80.30	87.12

Figure 1: Trained network and post-training regression analysis for [5-7-1] ANN Model.

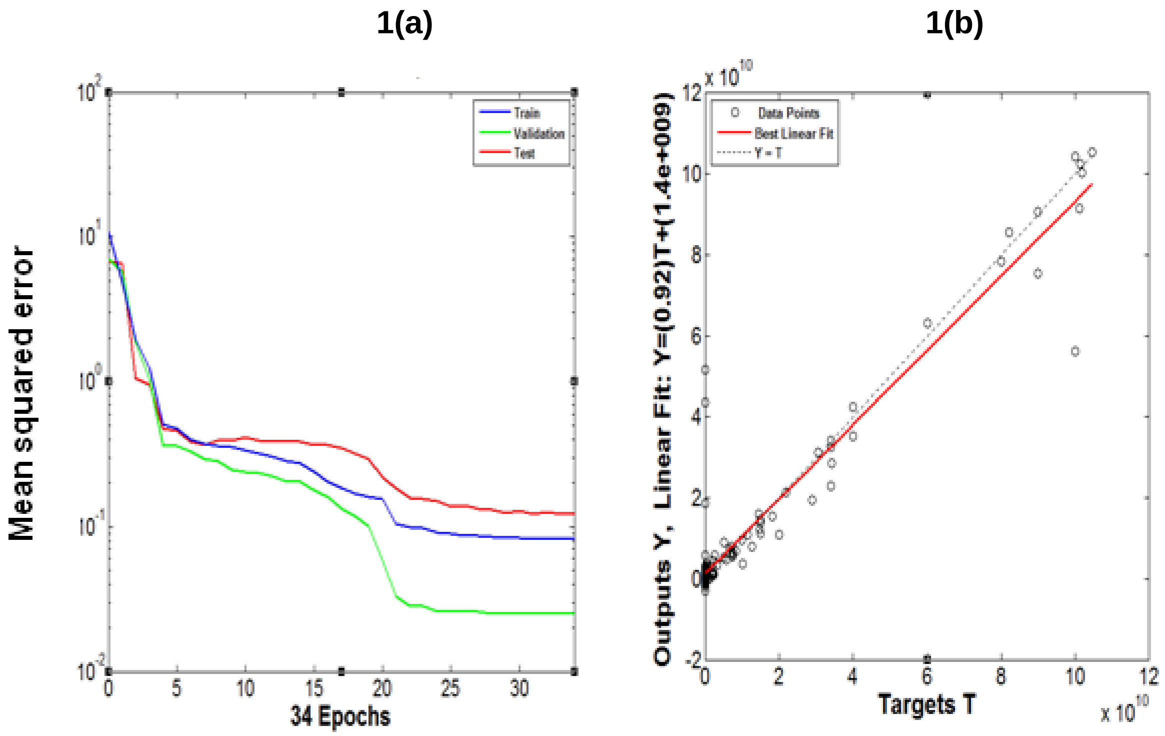
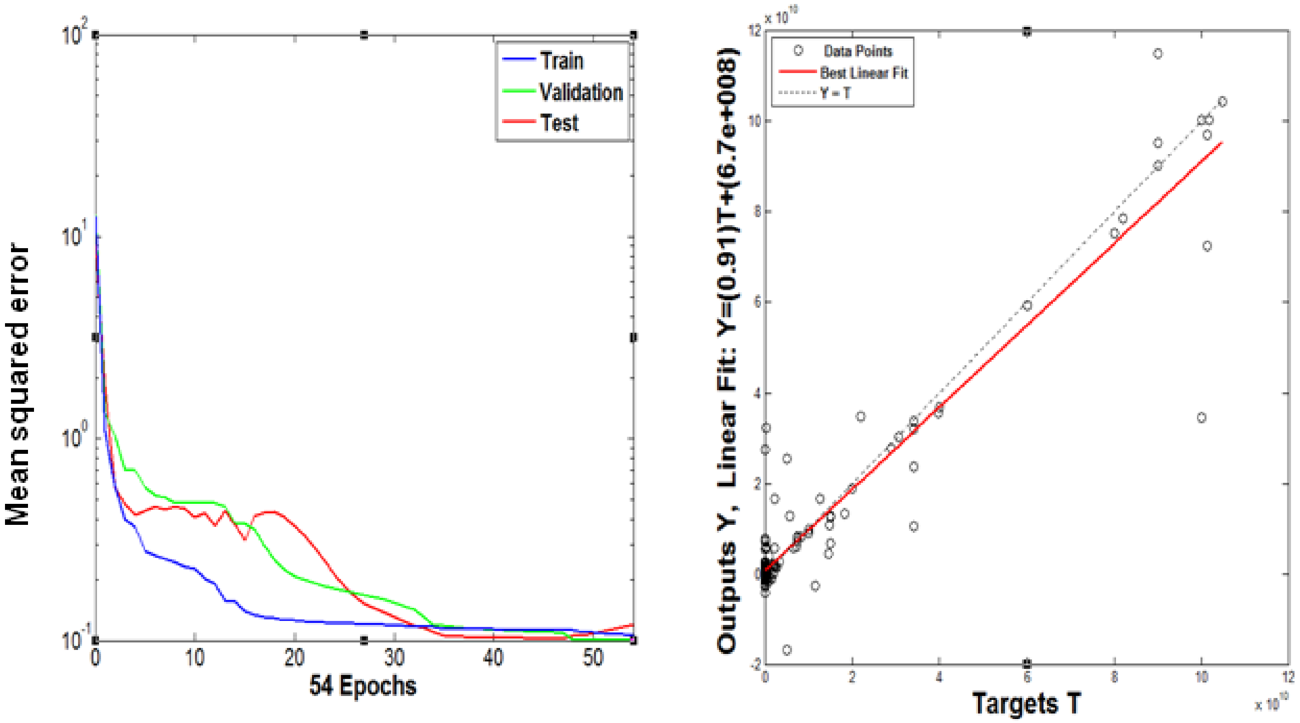


Figure 2: Trained network and post-training regression analysis for [5-9-1] ANN model.



2(a)

2(b)

Figure 3: Trained network and post-training regression analysis for [5-9-11-1] ANN model.

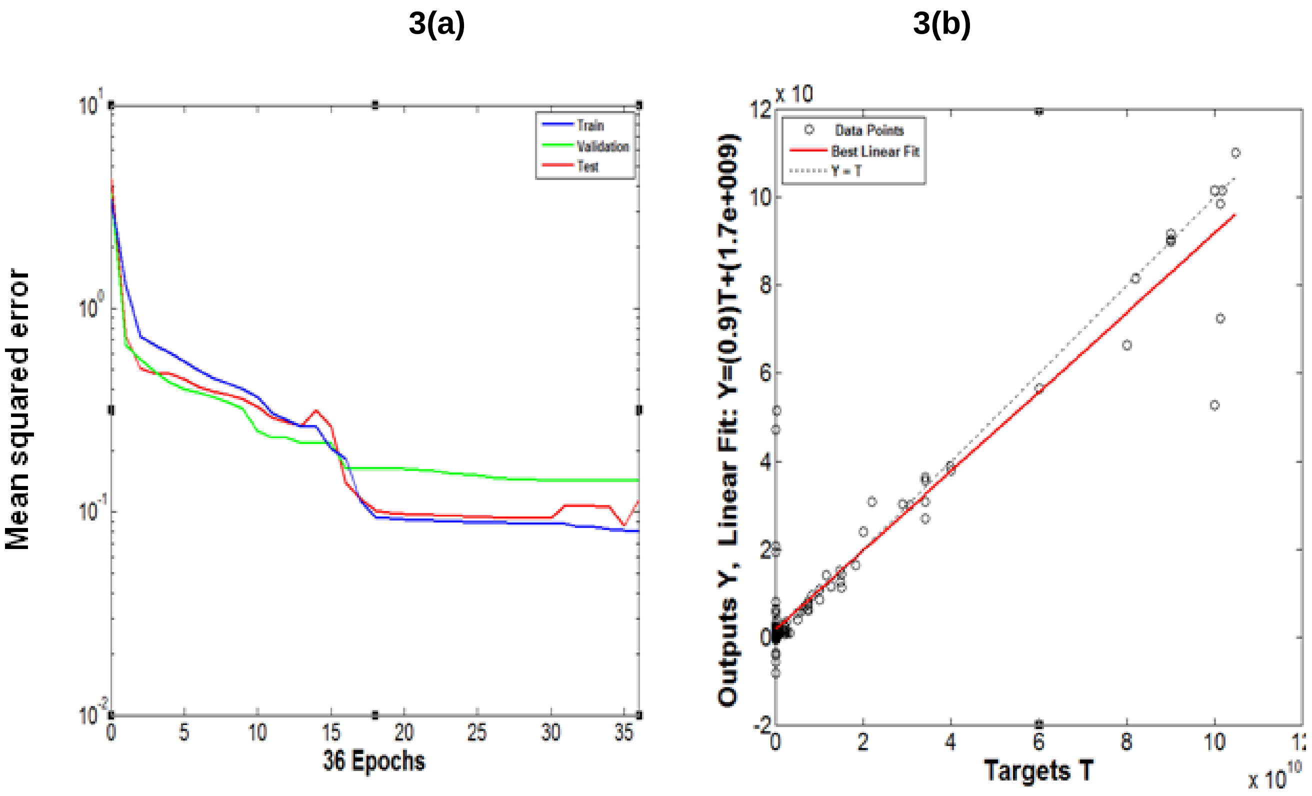
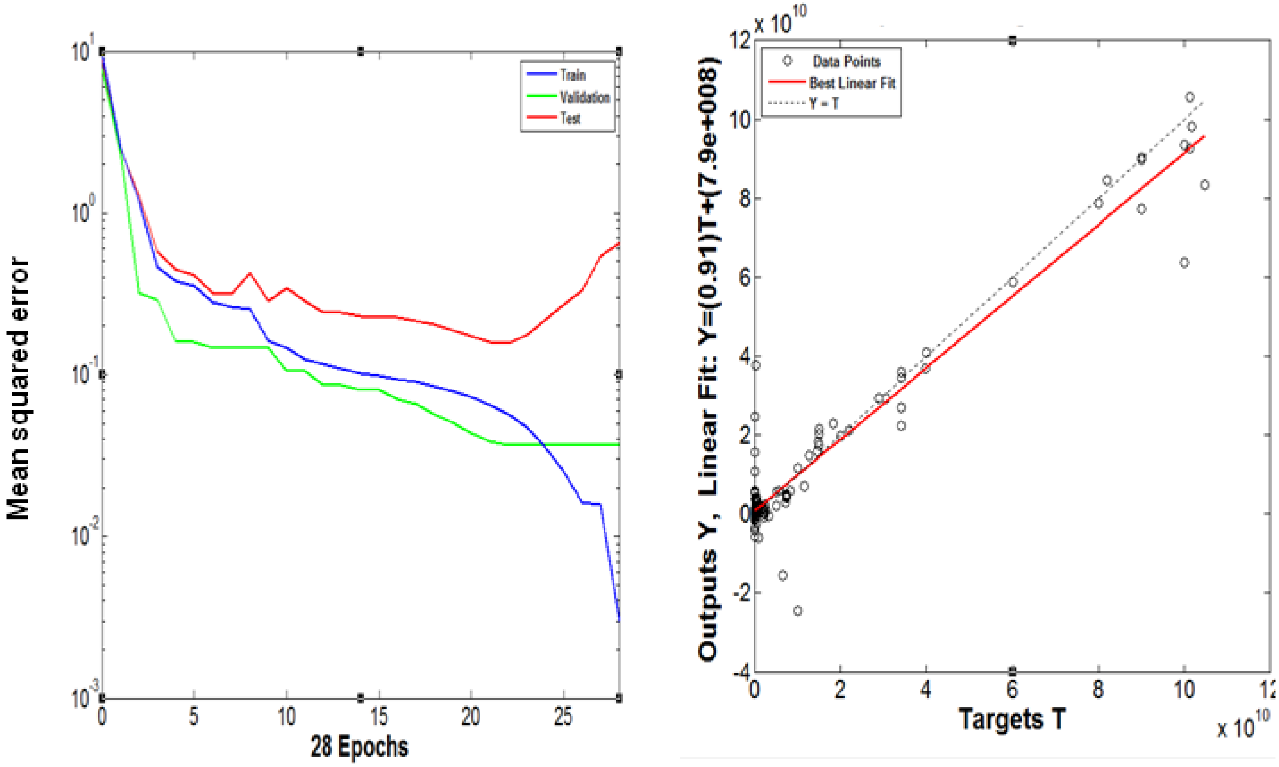


Figure 4: Trained Network and Post-Training Regression Analysis for [5-11-13-1] ANN Model



4(a)

4(b)

Figure 5: Average absolute relative errors (AARE): comparison for various regression (linear (LR) and non-liner (NLR)) and ANN models

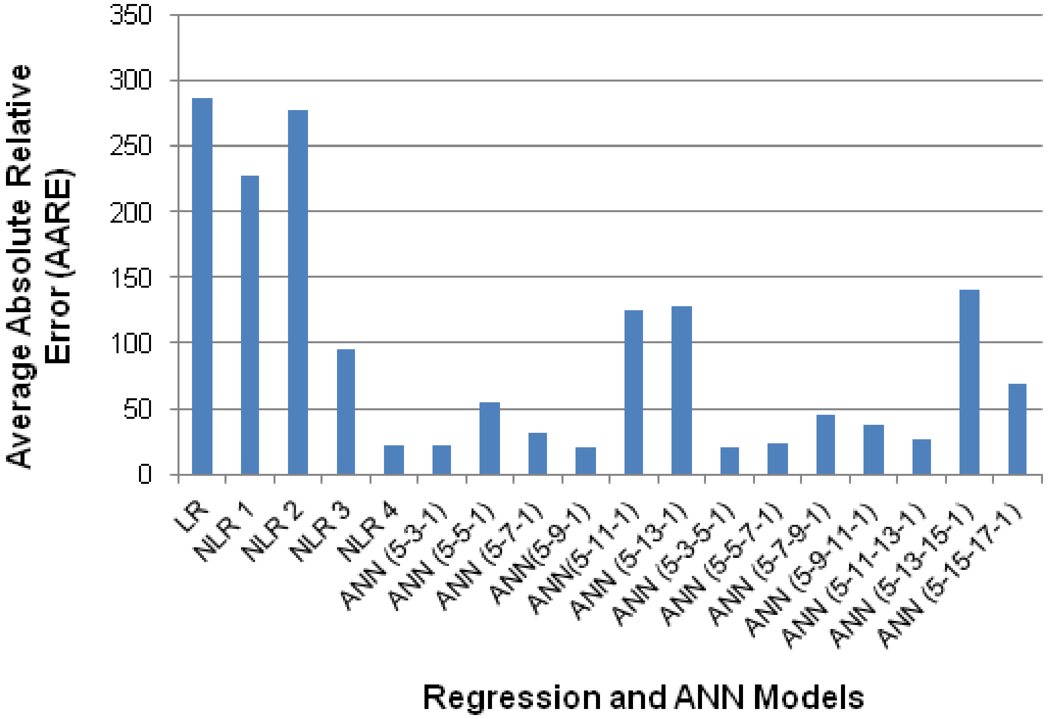


Figure 6: Sum squared errors (SSE): comparison for various regression (linear (LR) and non-liner (NLR)) and ANN models

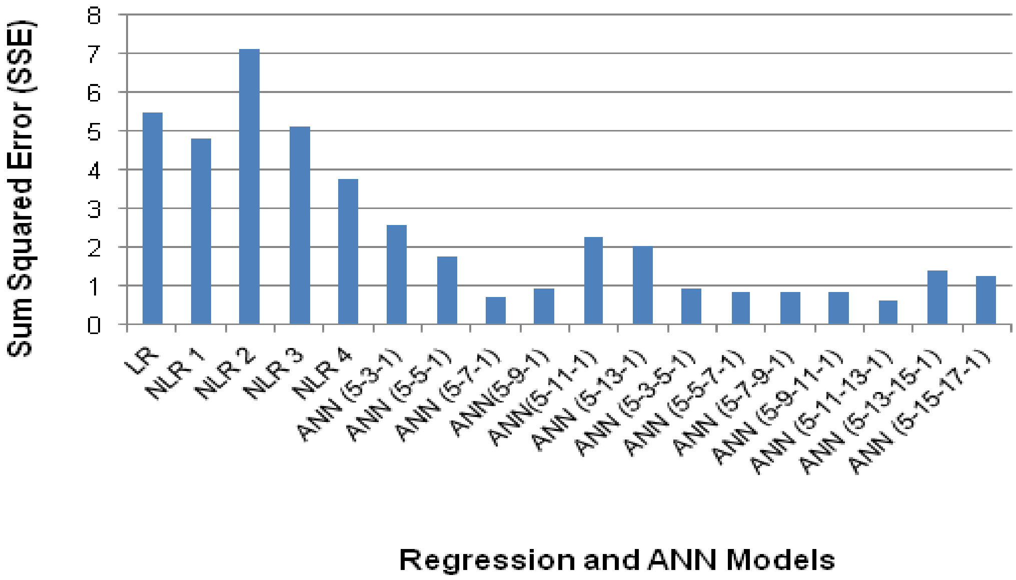


Figure 7: Efficiency (E) and Correlation Coefficient (R): comparison for various regression (linear (LR) and non-liner (NLR)) and ANN models

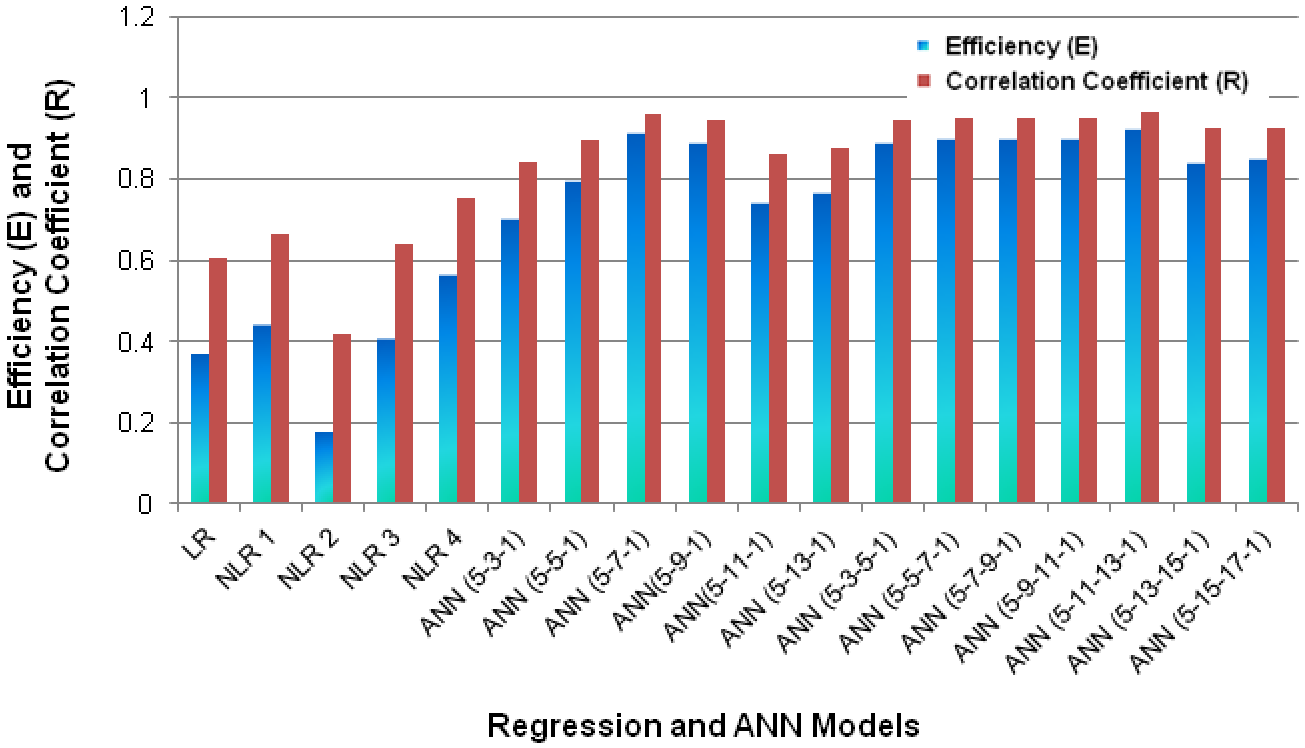


Figure 8: Threshold statistics (TS-5, TS-10, TS-25, TS-50 and TS-100): comparison for various regression (linear (LR) and non-liner (NLR)) and ANN models

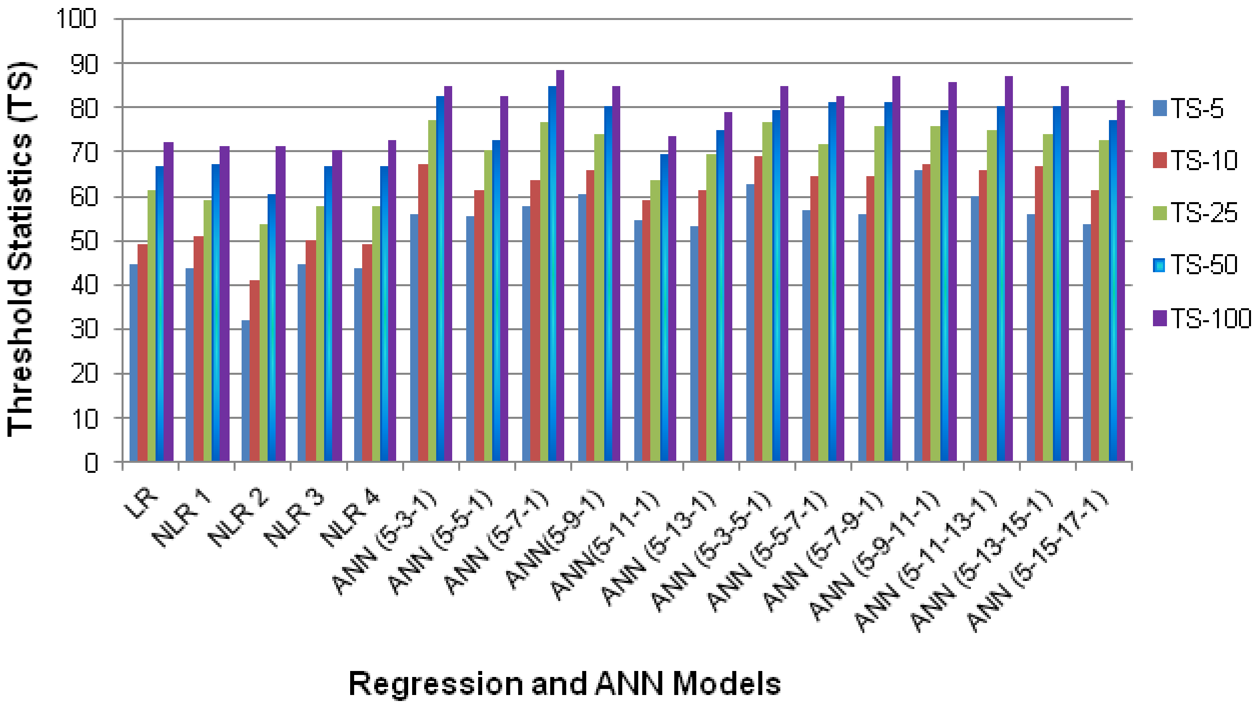
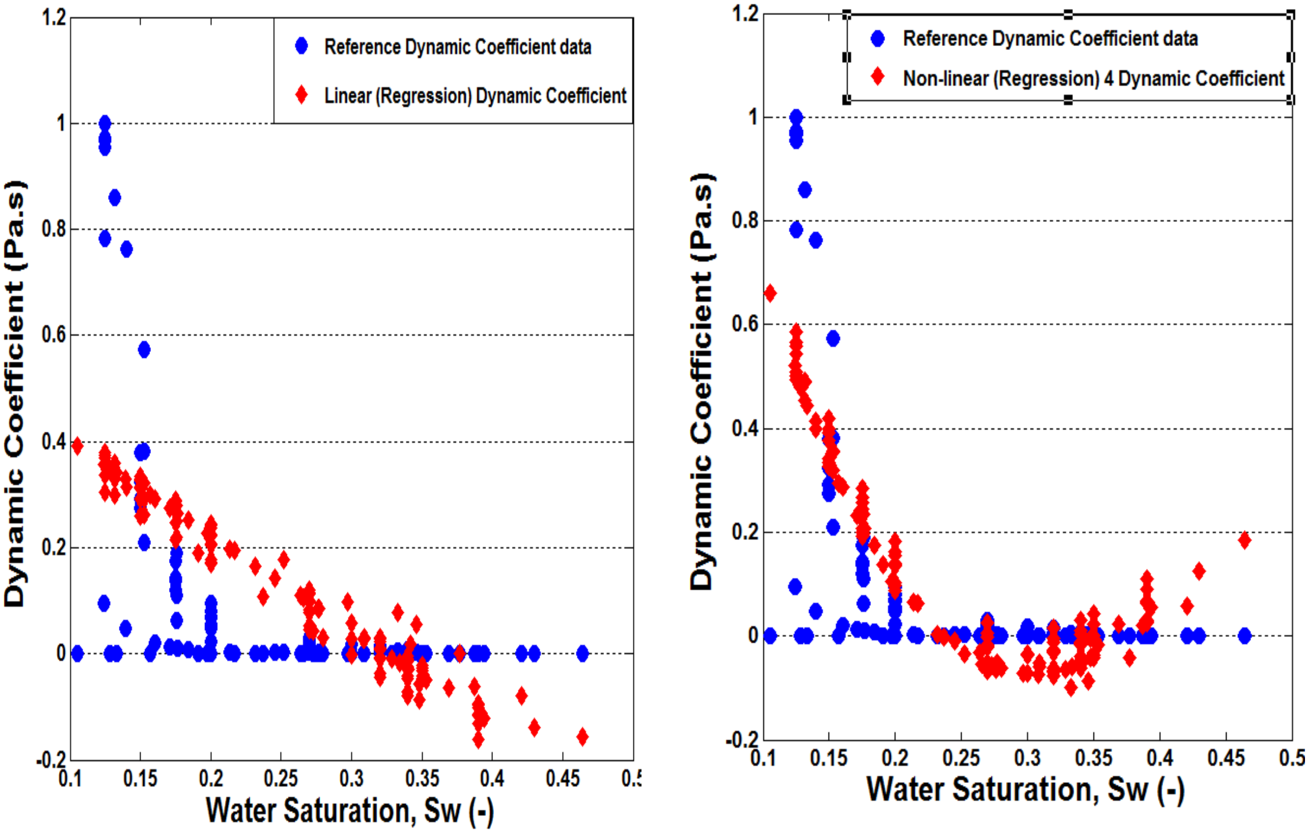


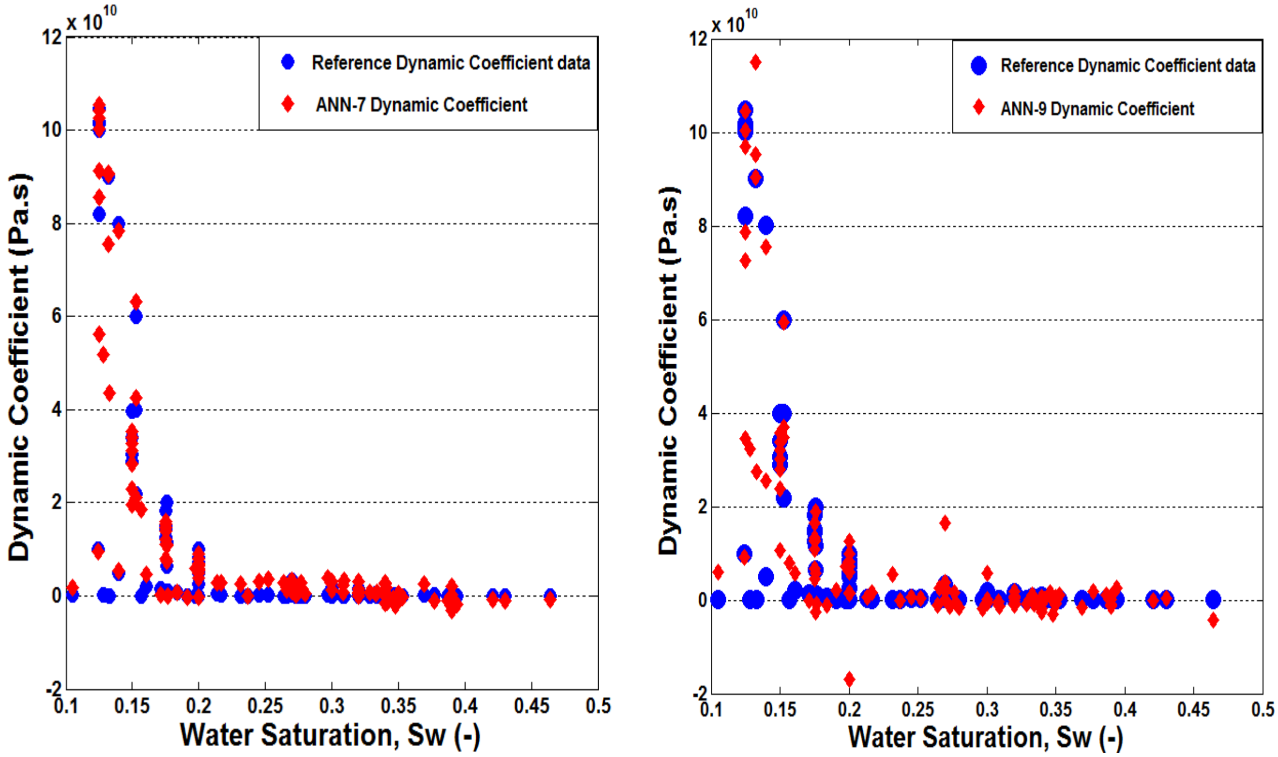
Figure 9: Regression model performance for predicting dynamic coefficient-water saturation relationship.



9(a)

9(b)

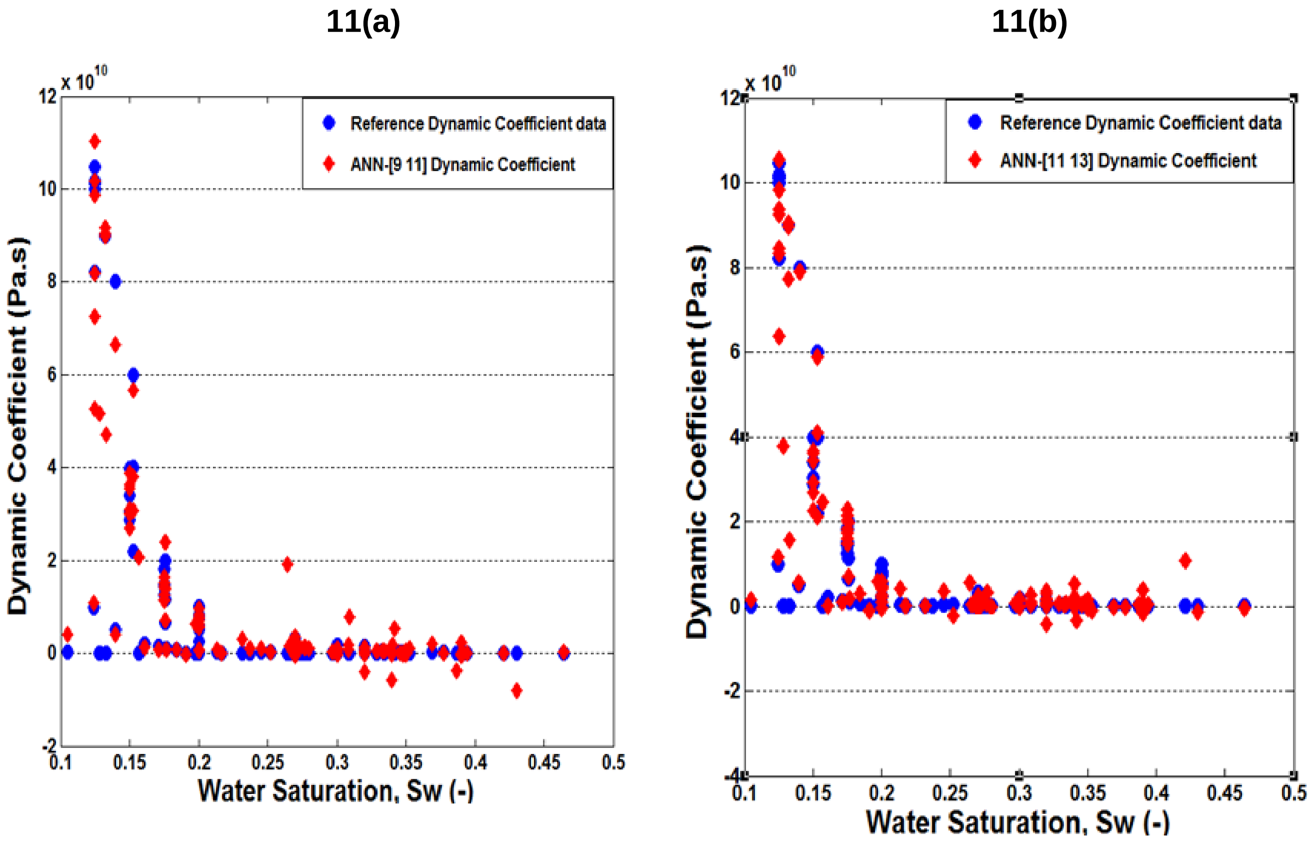
Figure 10: Single hidden layer ANN model performance for predicting dynamic coefficient-water saturation relationship. The ANN structure with [7] neurons in the hidden layer seem to perform better.



10(a)

10(b)

Figure 11: Double hidden layer ANN model performance for predicting dynamic coefficient-water saturation relationship. The ANN structure with [11 13] neurons in the hidden layers seem to perform better.


11(a)

11(b)

Figure 12: Comparisons of the best performing single and double hidden layer ANN models for predicting dynamic coefficient-water saturation relationship simulations. The performances of the two structures seem to be similar suggesting that one may choose single hidden layer ANN model

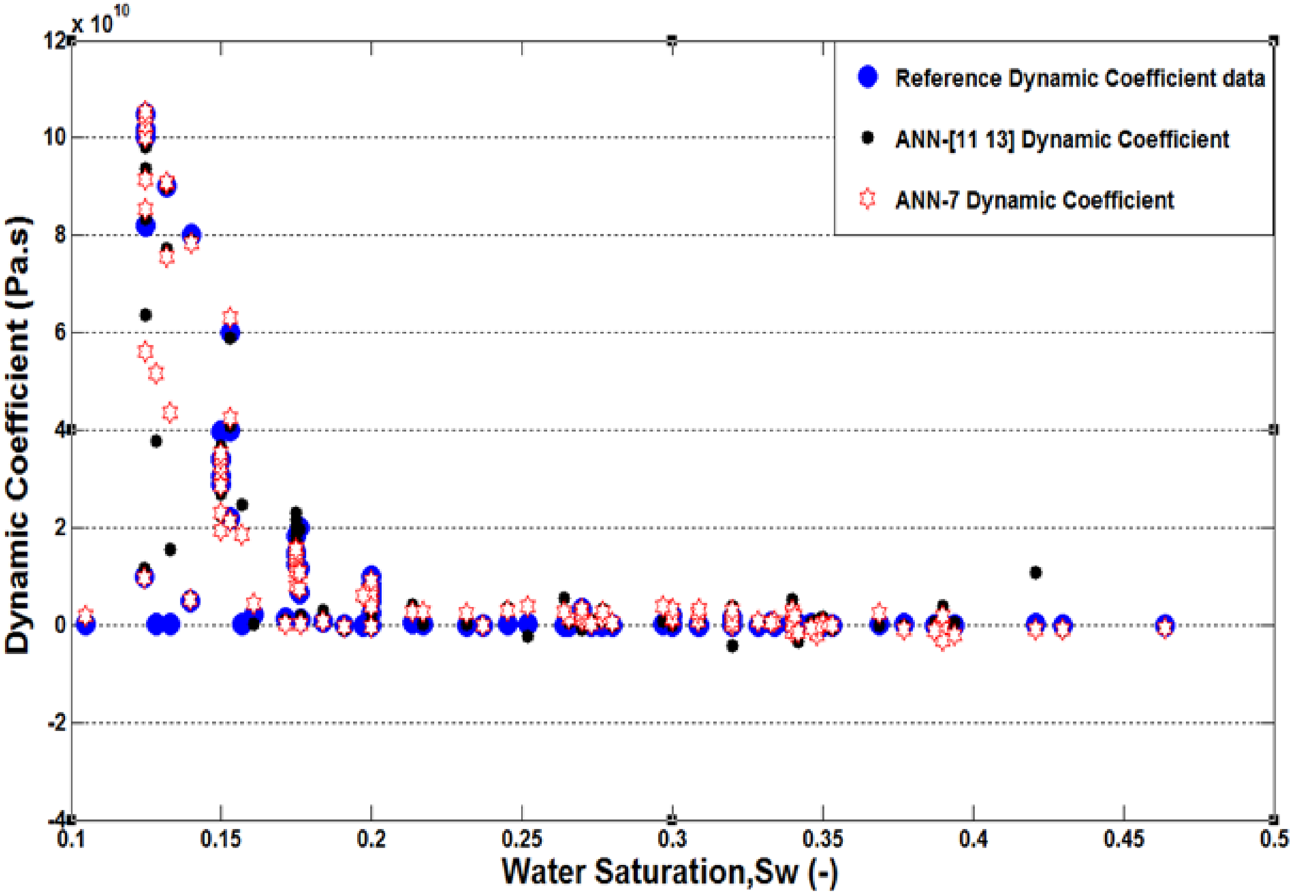


Figure 13: Best performing single Layer ANN model containing 5 inputs, 7 hidden layers and 1 output Neuron

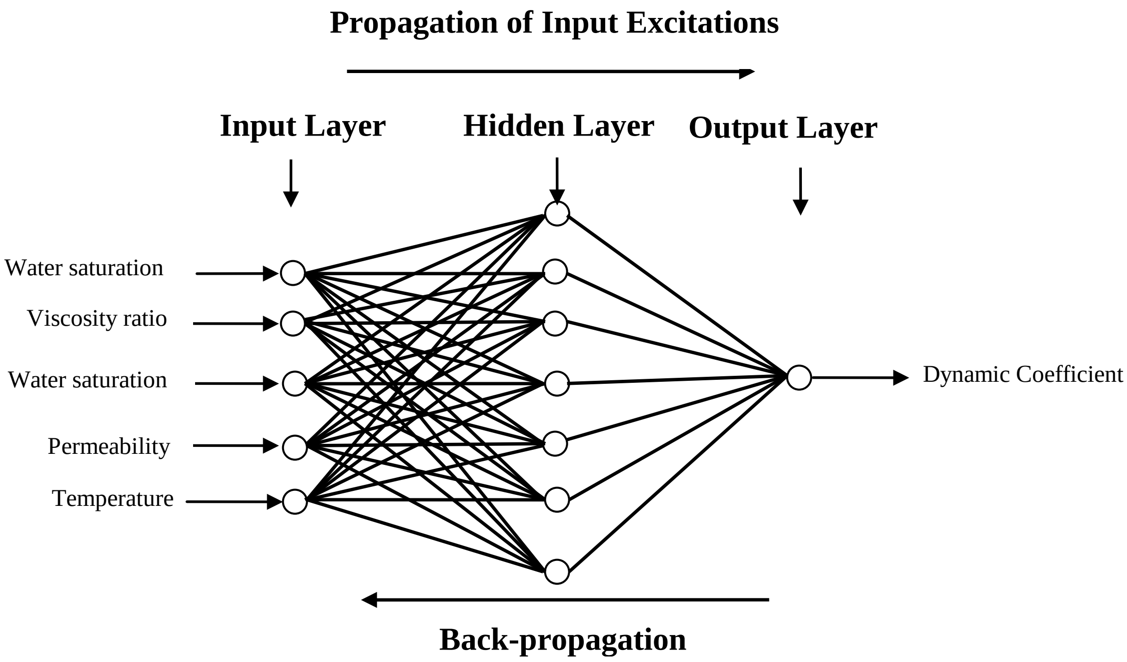
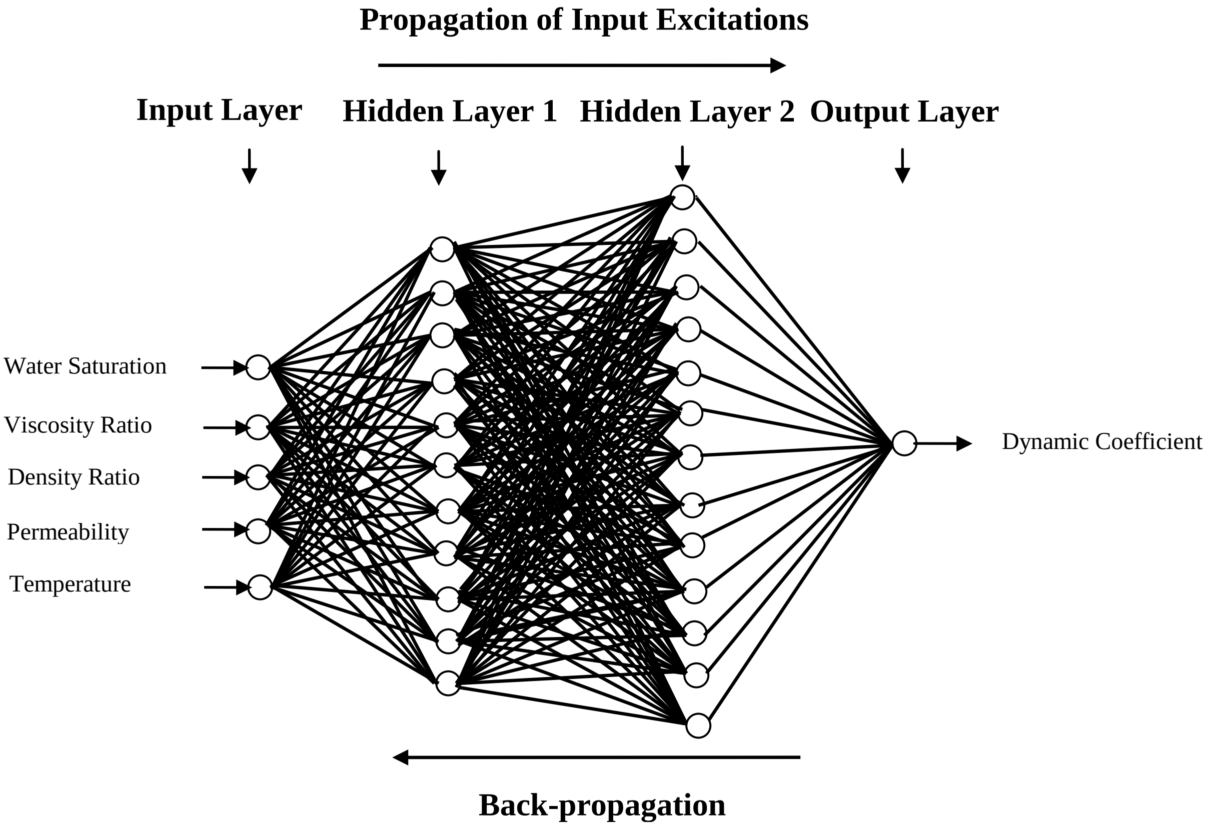


Figure 14: Best performing double Layer ANN Model containing 5 inputs, [11 13] hidden layer and 1 output Neuron


Propagation of Input Excitations
Input Layer
Hidden Layer 1
Hidden Layer 2
Output Layer
Water Saturation
Viscosity Ratio
Density Ratio
Permeability
Temperature
Dynamic Coefficient
Back-propagation