Prediction of Formation Damage During Fluid Injection into Fractured, Low Permeability Reservoirs via Neural Networks

M. Nikravesh, A.R. Kovscek, R.M. Johnston, and T.W. Patzek

1University of California at Berkeley, 2Lawrence Berkeley National Laboratory, 3CalResources, LLC
SPE Members

Abstract
The coupled, nonlinear and dynamic mechanisms that affect fluid injection for pressure maintenance or displacement, and oil production, are not well understood in low permeability fractured reservoirs. Thus, it is difficult to select an injection policy which maximizes oil recovery while minimizing formation damage caused by fluid injection and withdrawal. Here, we show that neural network models can be developed and used to predict, on a well-by-well basis, the dynamics of low permeability, fractured reservoirs undergoing fluid injection. The networks are trained using historical data from field operations.

We present an example from (i) a water and (ii) a steam injection project where over-pressurization has lead to unwanted extensions of fractures. First, using data from a waterflood project in the South Belridge Diatomite (Kern County, CA), we have built a neural network to predict wellhead pressure as a function of injection rate, and vice versa. The resulting model provides an excellent correlation between the inputs and outputs and recognizes major patterns in the input data structure, even though the behavior of the waterflood is complex. Second, using data from a dual injector streamdrive pilot in the same field, we have created neural networks which correlate the injection pressures and rates, and temperature responses in seven observation wells. Assuming a future injection pressure policy, the neural networks predict the injection rate and growth of heated reservoir volume. These predictions are then combined with a history-matching reservoir simulator to demonstrate how predictive simulation can be achieved even when mechanisms of steam injection and oil displacement into a tight fractured rock are not fully understood.

Introduction
Injection of water, CO2, or steam into low permeability fractured rocks such as diatomite, chalks, or carbonates for either pressure maintenance or oil displacement is problematic. On one hand, injection rates must be low enough to prevent reservoir damage from over-pressuring and inducing unwanted fractures. On the other hand, these rates must be high enough to make the costly fluid injection process economic. Historically, the conflict between prudent reservoir management and meeting injection targets has resulted in significant reservoir and well damage, injectant recirculation and irreversibly lost oil production. Much effort has been expended in recent years to develop models and theories for predicting tight rock behavior during fluid injection. However, the outcome has been less than satisfactory, and we still cannot tell reservoir engineers how to best produce low permeability, fractured reservoirs without incurring extensive formation damage.

Currently, reservoir engineers develop fluid injection policy on the basis of past experience, partial knowledge of the state of reservoir stress, production history, and limited predictions of future reservoir performance from numerical simulation. Neural network models for analyzing and predicting complex reservoir behavior are a promising new
engineering tool. Unlike other models, neural networks are capable of making accurate predictions even if all mechanisms affecting injection, production, and formation damage are not elucidated because these networks do not require specification of a structural relationship between input and output data.

The objective of this paper is to demonstrate the capability of neural networks to model reservoir behavior and to report the steps required to design a neural network model that predicts the dynamics of water or steam injection wells. Likewise, we discuss the frequency of wellhead data collection necessary for accurate modeling. We concentrate on the behavior of individual wells and discuss fieldwise prediction and management of injection in a companion paper [1].

Successful implementation of a neural network model requires extensive data sets, from pilots or computer assisted field operations, that sample a wide variety of reservoir behavior. We choose to analyze historical data from the South Belridge Diatomite (Kern County, CA) because both waterflood and steamdrive operations have been carried out there and the field is shallow, thereby permitting a high density of injection, production, and observation wells. Thus, we can verify the generalization properties of our neural networks for an entire spectrum of reservoir behavior. Although some properties of the diatomite are unique, it is an excellent analog of other deeper fractured reservoirs. Accordingly, the methodology developed here should be applicable to injection into other tight fractured reservoirs such as the Austin Chalk and the West Texas Carbonates.

California Diatomites. The diatomaceous oil fields of California, members of the prolific Monterey formation, are located in the San Joaquin Valley, west of Bakersfield, CA. The estimated original-oil-in-place in the California Diatomites exceeds 10 billion barrels and is comparable to that in Prudhoe Bay, Alaska [2]. They are largely undeveloped, but are relatively well characterized with the oil quantity and quality known. Although the Diatomites contain an astounding volume of oil, they also present severe engineering challenges. Matrix permeability is low, ranging between roughly 0.1 and 10 mD, while porosity is quite large, 25 to 65%. Further, the Diatomites have a high rock compressibility, but the rock is also naturally fractured. Natural fractures may be open or cemented shut. Additionally, diatomite is chemically unstable and exists as various forms of opal or quartz depending on depth and temperature [3]. It is possible to dissolve diatomite, transport it some distance, and then reprecipitate it as a different silica phase.

Diatomite reservoir architecture takes the form of a series of stacked silica rich layers with thicknesses ranging from a few inches to tens of feet separated by shales, silty clays or mudstones [3]. To compensate for low permeability and improve efficiency, both injectors and producers are hydrofractured. A typical well has 3 to 8 fractures with tip-to-tip fractures of about 300 ft [2]. Even after fracturing, primary recovery remains transient for many years because of low rock permeability. A typical oil recovery after 10 years on primary is 2.5 to 6.0%. Indefinite primary production is impossible to achieve because of reservoir compaction, subsidence, and severe well failures.

Waterfloods in the diatomite, implemented to sustain oil production and arrest subsidence, have suffered from low injectivity, poor vertical and areal sweep, severe extensions of injection hydrofractures, injector-producer linkage, and increased rates of well failure [4]. Every drop of secondary oil that is displaced during a waterflood must be contacted by water. Low rock permeability, low water injectivity, and large formation thickness all conspire to limit the success of waterflood in the diatomite. The best approach to improve waterfloodining is to carefully balance imbibition, water injection, and production so as to promote good volumetric sweep and stable water displacement fronts. This approach demands prediction and monitoring of waterflood performance at an unprecedented level.

On the other hand, steam can displace oil without contacting it directly. Oil heated by thermal conduction expands and evaporates, and is thereby expelled from the rock matrix. Hence, heat sweeps areas of the reservoir never directly contacted by steam. With steam injection in low permeability formations, success is predicated on maximizing heat delivery while minimizing the formation damage. To date, steam injection has not yet been applied commercially in the diatomite. Shell has conducted two limited steam drive pilots (Phase I and II) in the South Belridge Diatomite [5, 6], Mobil is conducting a heavy oil steam drive pilot in the South Belridge Diatomite [7], and Chevron has steam soaked producers at Cymric [8].

Both Shell pilots have demonstrated that steam can be injected into the diatomite, significant formation heating over the entire diatomite column can be achieved, and a significant oil production response to steam can occur. The reservoir behavior is
far from simple though. For instance, in the Phase I pilot, injectivity increased roughly 10-fold over the first six years of injection due to hydrofracture extension, opening and reconnection of natural fractures, and dissolution of the diatomite by steam. The Phase II pilot exhibited these characteristics, as well as highly unsymmetrical heating due to preferred steam convection channels. Therefore, means of selecting and controlling injection pressures and rates must be devised to either prevent or limit the abrupt and large extensions of hydrofractures.

**Neural Networks.** Both isothermal and thermal oil displacement processes exhibit inherently complex, nonlinear, time varying, and nonstationary behavior. During water or steam injection in the diatomite, there are several factors which cause such behavior: (1) changes in the rock matrix permeability; (2) extension of the existing fractures, creation of new fractures, and linking of fracture networks; (3) changes in the reservoir temperature; (4) changes in the oil viscosity; and (5) dissolution or creation of the gaseous phase.

Unfortunately, only linear and simple nonlinear reservoir behavior can be captured and analyzed with conventional statistical methods such as ordinary Least-Squares, Partial Least-Squares, and nonlinear Quadratic Partial Least-Squares. Neural network analysis, unlike regression, does not require specification of structural relationship between the input and output data. Cybenko [9] and Hunt-Nielsen [10] have shown that a neural network model can approximate any continuous nonlinear relation and generate complex decision regions for input-output mapping with useful generalization.

Neural networks have the potential to model reservoir behavior from nonlinear complex multidimensional field data and may find wide application in reservoir engineering. To date, they have been used in petroleum engineering mainly as tools for assisting in well test analysis and for well log analysis, c.f. [11-15].

Details regarding neural networks are available in the literature [9, 10, 16, 17]. Therefore, only the important network characteristics are mentioned here. The typical backpropagation neural network has an input layer, an output layer, and at least one hidden layer as illustrated in Fig. 1. There is no theoretical limit on the number of hidden layers, but, typically, there will be one. Each layer is fully connected to the succeeding layer with corresponding weights. The input-output mapping of the multilayer network shown in Fig. 1 can be represented by

\[ y_{\text{Net}}^{(Net)} = F_2 \left( W_2 y_{\text{Net}}^{(Net)} + b_2 \right) \]  

where

\[ y_{\text{Net}}^{(Net)} = F_1 \left( W_1 x + b_1 \right) \]

with

- \( n_x \): number of inputs
- \( n_h \): number of hidden layer nodes
- \( n_y \): number of outputs.
- \( W_1 \): \( n_h \times n_x \); input/hidden layer weight matrix
- \( W_2 \): \( n_y \times n_h \); hidden/output layer weight matrix
- \( b_1 \): \( n_h \times 1 \); hidden layer bias vector
- \( b_2 \): \( n_y \times 1 \); output layer bias vector
- \( y_{\text{Net}}^{(Net)} \): \( n_y \times 1 \); network prediction vector
- \( x \): \( n_x \times 1 \); network input vector
- \( y_{\text{Net}}^{(Net)} \): \( n_h \times 1 \); hidden layer output vector

The nonlinear transfer function, \( F_1 \), used in this work for all the network layers is the sigmoid hyperbolic tangent

\[ F_1(z) = \frac{e^z - e^{-z}}{e^z + e^{-z}} \]

As the neural network learns, the information is propagated back through the network and used to update the connection weights. Learning may require showing a network many thousands of examples. The objective function for the training algorithm is usually set-up as an optimization problem and is defined as the sum of errors squared,

\[ E = \frac{1}{2} \sum_{i=1}^{P} \left( y_{\text{observed}}^{(i)} - y_{\text{predicted}}^{(i)} \right)^2 \]

where P is the total number of separate data items used to train the network. This objective function defines a local error for the observed value at the output layer which is propagated back through the network. During learning or training of the network, the weights are adjusted to (i) minimize this sum of squared errors, (ii) improve the performance of the network, and (iii) provide the network with memory necessary in a learning process. Once neural networks are trained.
with information that spans a wide range of system behavior, they become excellent predictive tools. In addition, neural networks have the ability to infer general rules and extract typical patterns from specific examples, as well as to recognize input-output relationships from complex field data[18]. These properties give neural networks the ability to interpolate between typical patterns of data and generalize their learning in order to extrapolate to the region beyond their training domains.

In general, the performance of neural networks is a function of hidden layer topology. Useful generalization is affected by the number of hidden nodes and not by the number of hidden layers. In comparison with one hidden layer, two or more hidden layers do not significantly aid the recall process [19]. Therefore, depending on the complexity of the problem, useful generalization requires a minimum number of hidden nodes, but not a minimum number of hidden layers. In addition, the generalization results produced by multiple hidden layers and discriminatory capability with more than one hidden layer can lead to erroneous predictions. However, single hidden layer networks offer useful generalization and they generally train faster than multiple hidden layer networks. It can be shown that there exists a neural network with one hidden layer topology which has at least the same performance as a multi-hidden layer network [19]. Therefore, there seems to be no reason to use more than one hidden layer network in preference to a multi-hidden layer network in most applications. In addition, it has been shown [9, 10] that backpropagation networks with three layers can approximate any continuous nonlinear function and generate an accurate input-output mapping. Hence, only single hidden-layer networks are used here.

**Waterflood**

There are two approaches to analyze and predict the performance of waterfloods. In the first approach, the behavior of each injector or producer is considered independently and modeled with a neural network. For an injector or producer, historical data consisting of flow rate and well pressure are used along with the assumption that the same strategy of well operation will continue into the future. This approach uses a very simple neural network model and is easy to train and implement based on a minimum amount of information from the field.

In the second approach, the behavior of the waterflood is considered as a coupled, highly nonlinear system of injectors and producers. The field-wise objective is to meet a given production goal with the minimum amount of injectant. The oil field is divided into sections with similar characteristic behavior and each well within a section and its interaction with the other wells is studied. The model helps improve waterflood management and the design of recovery strategies. Our field-wise management approach is the subject of another paper [1] and here we concentrate on predicting the performance of single injectors.

**Model.** Figures 2 and 3 show typical behavior of a waterflood injector located in the South Belridge Diatomite. The well is undergoing a pressure-step test to judge injectivity. The injection rate and pressure data are noisy and occasionally miss information. Actual data are displayed with dashed lines and represent one-hour averages of measurements acquired every second by a Computer-Assisted Operations (CAO) system for Shell's South Belridge diatomite waterfloods.

The design of a neural network to predict the behavior of an injector starts with filtering, smoothing, and interpolating values for missing information in the historical data set. A first order digital or analog filter and a simple linear recursive parameter estimator [20, 21] for interpolating is all that is needed to filter and reconstruct the noisy data. Figures 2 and 3 show the performance of the filtering, smoothing, and reconstruction operation on the wellhead pressure as a dark solid line.

Our model to predict the behavior of an injector has 3 input nodes representing the current scaled values of pressure, current scaled injection rate, and the scaled step change in pressure. All inputs are scaled between 0 and 1 using the maximum rates and pressures for a particular well. The hidden layer contains 10 nodes. The output layer has 75 nodes representing the prediction of the injection rates.

Initially we trained the network using a backpropagation algorithm and the reconstructed data shown in Figs. 2 and 3. The subsets of the historical data marked by the numbers 1, 2, and 4 were presented to the network for training purposes. Training continued until we found that the network prediction suffered upon continued training. Data sets 3 and 5 were used later for interpolation and extrapolation to test the network performance. This network accurately predicts the behavior of the reservoir 75 time steps into the future, where each step represents an interval of 2 hours. For predicting outputs more than 75 time-steps into the future, iteration through the neural network would be required. Hence, the
predicted outputs from the network are reused as inputs, and the inputs are shifted accordingly.

Figures 4 through 8 show the performance of the neural network model in more detail. The network models very well the training data sets 1, 2 and 4, as shown in Figs. 4, 5 and 6, respectively. The thick dark line gives the input wellhead pressure to the network. The actual water injection rate is given as a solid line, while the neural network prediction of injection rate as a function of wellhead pressure is given by a dashed line.

To demonstrate the generalization properties of our network, we used it as both an interpolator and extrapolator. Wellhead pressures from data sets 3 and 5 were presented to the network and water injection rate was predicted using the network weights and biases found from training it with data sets 1, 2, and 4. The performance of the network for interpolation (data set 3), and extrapolation (data set 5) is shown in Figs. 7 and 8. As we can see, this neural network maps well the inputs (injection pressures) onto the outputs (injection rates) and serves as an excellent interpolator and extrapolator. Thus, the neural network model has good generalization properties.

Figures 3 through 8 show typical results from our study of waterflood behavior in the South Belridge Diatomite. It is important to note that this study used a minimum amount of field information, i.e., only injection rates and injection pressures. It is possible to introduce more information into the network model to constrain it. For example, a rock mechanics model could be used to predict extensions of a hydrofracture. An estimate of the hydrofracture location, orientation and size might be used to predict injector-producer linkage, etc.

Most of the wells examined in this field showed only a single fracture extension. If the fracture extension data are introduced to the network during the training period, the network captures this extension. However, a perfect match of a past fracture extension is no guarantee that the network will accurately predict future fracture extensions. This is because at present there are no other fracture extensions in the well that may be used to retrain the network. A predictive rock-mechanical model used in conjunction with a neural network should remedy this potential deficiency.

To show that it is possible to predict fracture extensions using the neural network model, we studied a more complex situation. First, a part of the waterflooded field was chosen. The network was then trained based on known information from wells in that part. Included in the training set were the fracture extension behavior and normal behavior of several injectors. Hence, the neural network accumulated information from several wells. The network was then used to predict the performance of another well within the same part of the field. This particular well had never been shown to the network. Figure 9 shows the wellhead pressure as a function of time, while Fig. 10 shows the behavior of the test injector. Data to the left of the dark, vertical, dashed line at roughly 12 days are introduced to the previously developed network model for additional training. It is important to note that this small amount of known information is not a sufficient data set if used as the sole source of training information. Figure 11 shows the network prediction of water injection rate using wellhead pressure as input. Comparing Figs. 10 and 11, it is evident that the network is able to accurately predict injection rate and capture the dynamics of the hydrofracture extension that occurs at roughly 64 days. Hydrofracture extension is marked by a large increase in injection rate over a short period of time.

Since the network has learned the symptoms of injection leading to hydrofracture extension, it can be used to suggest an injection policy for the well that minimizes formation damage. Based on the knowledge contained in the weights and biases, the network model suggests that wellhead pressure not be allowed to exceed 150 psi. In Fig. 9, the actual injection pressure is stepped to approximately 160 psi at 64 days and to about 170 psi at 75 days. The lighter line between 64 and 90 days in Fig. 11 shows the network prediction of injection rate based on the more conservative policy. The network predicted injection rate is low because a wellhead pressure of 150 psi is not high enough to induce a hydrofracture to extend based on the historical data from wells in this part of the field.

Figure 12 replots the injection data for this example as the cumulative injected water as a function of time. The solid dark line is the actual field performance, and the gray line is the neural network prediction of the actual performance. Lying much below these two curves is the result of the injection policy suggested by the neural network. Because the wellhead pressure is limited, no fracture extension occurs and the cumulative injection continues along the same trend. This result is consistent with our goal of maximizing oil recovery while minimizing formation damage and fluid injection.

**Fracture Extension**

After reviewing waterflood dynamics in the South Belridge Diatomite, we have learned that, historically, an important factor causing fracture extension is the
aggressive action of Proportional-Integral-Derivative (PID) controllers during start-up periods, or when the system is near the fracturing gradient. The effects of aggressive controller behavior are evident upon comparing Figs. 2 and 3 with Figs. 9 and 10. The behavior of both injectors is very similar as fracture extension occurs in both wells when the wellhead pressure approaches 150 psi.

However, there are important differences between these two injectors. Figures 2 and 3 show that the behavior of the reservoir around the well after the first fracture extension (point 5) does not differ substantially from the behavior before the fracture extension. Injection response to changes in the wellhead pressure, both before and after hydrofracture extension, displays a classical square root of time decline in the rate of injection [4]. This is not true for the second well shown on Figs. 9 and 10. After hydrofracture extension, injection rates show no decline upon establishing a new wellhead pressure.

Both cases are examples of forced fracture extension. In response to aggressive PID action which creates a water hammer in the wellbore, the fracture opens for a period of time allowing large water injection rates even though the final pressure does not lead to a pressure gradient above the fracturing gradient. Essentially, the fracture fills with liquid as a result of injection, the liquid causes the fracture to extend, and then the liquid squirts into the formation in response to a pressure perturbation. It is important to note that the data available for this study were six hour averages of well behavior. Since we believe that the dynamics of forced fracture extension occur over a much shorter time period than 1 hour, the actual behavior of the PID controller during fracture opening is not shown by these figures. Thus for a time period shorter than the 1 hour average, the pressure gradient exceeds the fracturing gradient. The length of the period of time during which the reservoir stays equal to or above the fracturing gradient, determines whether fracture extension is temporary or permanent. Figures 3 and 4 demonstrate a temporary fracture extension in that the system returns to it pre-extension behavior, while Figs. 10 and 11 display a permanent fracture extension.

There are several ways to prevent such fracture extensions. One is to retune the PID controller to improve performance and reduce controller aggressiveness. In general, retuning PID controllers is time consuming and requires a combination of operational experience and trial-and-error procedures. A neural network model can assist greatly in retuning by learning the good and bad symptoms of the PID controller behavior and then suggesting new controller setpoints and gains. Figures 9 through 11 show that based on what the network learned, it was able to suggest a better operating procedure. Aggressive PID action was prevented, thereby preventing fracture extension.

A second approach is to use the network as a model identifier to assist the PID control. During the past few years, neural networks have been applied effectively as controllers for time varying processes with highly nonlinear behavior. It has been shown that the neural network model based control strategies are robust enough to perform well over a wide range of operating conditions, and they are much easier to design and implement than classical PID control [17]. Currently, we are developing a neural network model based control strategy for water and steam injectors to augment the present PID control strategy.

Steamdrive Behavior

Thus far we have only examined neural network models for the analysis and prediction of single well behavior. However, when used in conjunction with a first principles model, such as a reservoir simulation model, neural networks allow us to achieve predictive simulations while oil displacement and formation damage mechanisms are being explored.

As an example, we use neural network predictions of steam injection rate and temperature profiles in observation wells for the Phase II steamdrive pilot as inputs to a history matching simulator. The simulator functions by using temperature response and cumulative steam injection to infer the portions of a hydrofracture which conduct steam to the formation, the temperature distribution within the formation, and relative changes in matrix permeability [22, 23]. Unfortunately, this simulator is not predictive. Although our model can capture changes in matrix permeability, mechanisms for permeability evolution are not included in it. To be fair, predictive models for fracture extensions and formation plugging by silica precipitation are not parts of more sophisticated commercial simulators either.

Details of this pilot can be found elsewhere [5, 6]. In short, steam is injected through two hydrofractured injection wells, IN2U and IN2L, that span the entire Diatomite column at South Belridge and oil is recovered at two production wells lying to the northwest, 543N, and southeast, 543P, of the injection hydrofractures. IN2U is perforated from 1110 to 1460 ft, while IN2L is perforated from 1560
to 1910 ft; hence, there is no communication between the injection hydrofractures. Heating of the Diatomite by steam is measured in 7 observation wells that are distributed across the pilot area. Figure 13 gives a plan view of the pilot, the surface locations of the wells, as well as the individual names of the observation wells.

The hybrid reservoir simulation-neural network approach for predicting results of the Phase II Pilot functions in the following manner. Given steam injection rate, wellhead pressure, and the temperature responses at the observation wells for a given period of time, say 0 to 700 days of steam injection, a neural network model and the steam injection history matching simulator are run in parallel to obtain a best fit of the Phase II results. In the case of the neural network model, this entails predicting the change in temperature at each observation well for a given time interval.

The trained neural network is then used to extrapolate into the future the temperature response in each observation well and the steam injection rate. The temperature response and the cumulative injected steam predicted by the neural network are used in place of actual data as input to the history matching simulator. The simulator is then restarted with the output from the first 650 days of steam injection as initial conditions and “history matching” of the neural network data is performed.

Hence, this combination of neural network extrapolation of steam-injection response in time and history matching allows us to predict volumetric heating of the diatomite and the zones with the largest steam flow. As a sidenote, the power of neural networks to interpolate between complex nonlinear data also allows us to generate various injection scenarios and visualize their results [1]. Details of the neural network prediction are described prior to displaying the results of this hybrid approach.

**Neural Network Model for Steam Injection**

Figures 14 and 15 show typical behavior of the IN2L steam injector from the Phase II Pilot. The dashed lines representing the injection rate and wellhead pressure include substantial noise. Similar to the water injectors, the first step in designing a neural network model was to filter and smooth the actual injection data. Next, a series of neural network models for short term prediction of injection behavior were developed.

First, steam injection rate as a function of wellhead pressure is predicted, and then the so-called “inverse problem,” wellhead pressure as a function of steam injection rate, is predicted. The structure of each neural net is similar. Each model has 6 input nodes, 15 nodes in the hidden layer with nonlinear transfer functions, and 10 nodes in the output layer with nonlinear transfer functions. Network output is a prediction of either the injection rate or the wellhead pressure for the next 10 sampling periods. All rates and pressures are scaled using historical maxima in the data set. The sampling period between known values of wellhead pressure and injection rate is 1 day.

For prediction of steam injection rate the input nodes represent the current wellhead pressure, current injection rate, and 4 past values of wellhead pressure. The six input nodes for the network that predicts injection pressure represent the current injection rate, current injection pressure, and the 4 past values of the injection rate;

Each network is initially trained using a backpropagation algorithm with smoothed and reconstructed data. A moving window is used for training on the first 200 days of data, and the next 100 days of the data set are used for testing the network. The network is trained and updated in the next training window data set based on a simultaneous updating approach [17]. The training data sets labeled in Figs. 14 and 15 were presented to the network, and training was stopped when it was found that the network's prediction suffered upon continued training. Next, steam injection rate and wellhead pressure were predicted by each network, respectively.

The network has excellent prediction for the training data sets, as expected. To show the performance and generality of the network, the model was used for interpolation and extrapolation of injection 3 months into the future. The performance of the network for the test data is marked as a solid line in Figs. 14 and 15. Both figures illustrate that each neural network maps well inputs onto outputs and is a good interpolator and extrapolator. Thus, these neural network models have excellent generalization properties.

For network prediction of injection rate more than 10 time steps into the future, iteration through the neural network is required. Therefore, the predicted output from the network is used as input to the neural network and the input is shifted accordingly. It is important to note that the prediction will deviate gradually from the actual value as the network is iterated. For our model, as long as discontinuous changes in the behavior of the reservoir do not occur (e.g., an extension of the injection hydrofracture), the prediction is reasonable.

Also, note that even though the performance of the networks was perfect for this case study, a comparable performance is not guaranteed for other
cases. Therefore, we are conducting a more detailed study of modeling reservoir behavior as well as a model for field-wide management [1].

**Long Term Prediction of Steam Injection.** Since short term prediction of steam injection behavior was successful, we developed a neural network model for longer term prediction of steam injection rate as a function of injection pressure. This model is used to estimate the long term performance of steam injection in conjunction with a reservoir simulator. Using this network model, the results from different injection scenarios can be predicted allowing us to analyze and choose an optimal scenario. The model has 5 input nodes representing the current and 4 past, scaled values of the injection pressure, 20 nodes in the hidden layer, and 2 nodes in the output layer. The output gives a prediction of injection rates two sampling periods into the future. The model is called a feedforward model, because the injection rate is predicted solely from the known pressure history. The network was trained using a backpropagation algorithm with the reconstructed data given in Figs. 14 and 15. Training was stopped when it was found that the network's prediction suffered upon continued training. After the injection rate versus time relationship is known, calculating cumulative steam injection versus time is trivial.

For long term prediction, the error in the cumulative sum of the injection rate is backpropagated through the network for adjusting the weight and bias terms instead of the error in the daily injection rate. The performance of this model on a daily basis is not as accurate as the previous one. However, because the current model is constrained by the total amount of injected steam, its long term prediction of cumulative steam injection is excellent. In addition, the models for short term prediction may be used to assist the feedforward, long term model. Therefore, the combination of feedforward and the previous model for short term prediction can be used for better long term prediction if information is needed on a daily basis.

Figure 16 shows the performance of the network model for predicting the cumulative amount of steam injected versus the actual data. The network has good performance. Also shown are network predictions of cumulative injection given ± 10% changes in the wellhead pressure, thereby allowing us to see the effect of changes in the injection pressure on the cumulative steam injected. We conclude that by using simple neural network models, it is possible to predict the behavior of the injectors for both a short and long time periods to a reasonable degree of accuracy.

**Neural Network Model for Observation Wells.** The temperature responses in the observation wells displayed in Fig. 13 quantify the extent and uniformity of heating of the reservoir and the rate of expansion of the heated rock volume. The latter rate is a function of the injection pressure, the hydrofracture area and the creation of steam-flow channels in the formation. Hence, prediction of the future temperature responses is crucial to ensuring smooth operation of injectors and the prevention of unwanted fracture extensions.

Briefly, we have developed several neural network models to predict the temperature responses of the observation wells. A typical network has 3 input nodes representing the current and 2 past scaled values of temperature response, 2 nodes in the hidden layer, and 1 node in the output layer that gives a prediction of the temperature response 30 days into the future. The network is trained using a backpropagation algorithm. Figures 17, 18, and 19 show the typical temperature responses of Wells LO13, LO14, and LO15. Actual temperature measurements are shown as dashed lines, while the neural network predictions are given as solid diamonds connected by straight lines. Only the temperature responses at depths corresponding to the midpoints of the reservoir simulation layers to follow are shown, suggesting an imperfect match of the field data. However, our neural network models the temperature profiles exactly. Although not displayed, temperature responses at observation wells, LO11, LO12, MO1 and MO2 were generated with identical networks. Model predictions can then be used in conjunction with any reservoir simulator for further detailed studies of the reservoir under steam injection. Here we use only our history matching simulator with simplified physics to show that neural networks and reservoir simulators can be used in concert.

**Hybrid First Principle-Neural Network Model.** Figure 20 displays a plot of wellhead steam injection pressure for well IN2L as a function of time. The dark, dashed line gives the actual history from the pilot for the first 700 days of injection while the solid line gives the pressure input to the history matching simulator. Given the continuation of steam injection between 700 and 1200 days at the pressure indicated on Fig. 20, the neural networks are used to predict the temperature responses in the observation wells, in
addition to the cumulative injected steam. These predictions become the inputs to our history matching simulator to estimate the volumetric distribution of heat within the pilot and diagnose future performance.

As a demonstration of the results from this hybrid approach, Fig. 16 superimposes the simulator-predicted cumulative steam injection over the actual and the neural-network-predicted injection. Additionally, Figs. 17 to 19 show the neural network and simulator predictions of temperature response between 700 and 1200 days for observation wells LO13, LO14, and LO15. As shown on Fig. 13, these wells are adjacent to IN2L. The dark dashed line gives the actual temperature response between 1500 and 2000 ft. Neural network predicted temperature response is represented by solid diamonds and the simulator predictions of temperature response are marked by solid circles. The horizontal dashed lines and the letters J through M indicate the geologic layering at South Belridge.

To the east of IN2L, Fig. 19 displays a dramatic temperature response at a depth of roughly 1800 ft. This indicates that continuing the current injection policy will lead to dramatic temperature increases in LO15 and steam breakthrough at the close by producer, 543P. In fact, this was observed in the pilot [22, 23]. To the west of IN2L, Figs. 17 and 18 show vertical asymmetry of heating, but no acceleration of the temperature response.

**Summary**

Neural network models can match and then predict complicated reservoir behavior when historical databases are available. To capture detailed extensions of injection hydrofractures in tight rocks, the historical data must be acquired at 30-60 second intervals. Modern Computer-Assisted Operations (CAO) systems, mass data storage devices, and fast data transfer protocols provide the foundation for rapid data acquisition. With current computer hardware and networks, these requirements can be satisfied at a relatively low cost and the potential savings in terms of otherwise forfeited oil production may be huge. To capture large and permanent fracture extensions a much lower frequency of data acquisition, say 1 measurement per day, is required. The neural networks are capable of making accurate predictions even when all mechanisms affecting injection or production behavior are not known. Further, neural networks provide a way to incorporate disparate information because a structural relationship between input and output data is not required.

Using steam and water injection data from field operations in the South Belridge Diatomite, we have demonstrated that neural networks are capable of predicting injection rate as a function of wellhead pressure and vice versa. The networks used are simple and are based on accepted neural network designs and training algorithms. These networks match field data very well and have exceptional generalization properties. They are able to accurately predict extensions of injection hydrofractures and provide us with a means of preventing unwanted fracturing.

With regard to steam injection, neural networks used in conjunction with reservoir simulation provide a novel tool for predicting, visualizing, and screening various steam injection strategies. Thus, predictive simulation can be achieved several months into the future even when mechanisms of injection and oil displacement are not fully understood.

**Nomenclature**

- $E$ = objective function
- $F$ = transfer function
- $n$ = number
- $W$ = weight
- $x$ = network input
- $y$ = network prediction or hidden layer output
- $\theta$ = bias vector

**Subscripts**

- $h$ = hidden layer
- $x$ = inputs
- $y$ = outputs

**Acknowledgments**

We thank CalResources LLC, formerly Shell Western E&P, for providing the water and steam injector data. This work was supported by the Office of Laboratory Technology Applications, Office of Energy Research of the U.S. Department of Energy under contract No. DE-AC03-76FS00098 to the Ernest Orlando Lawrence Berkeley National Laboratory of the University of California.

**References**


Fig. 1 - Typical neural network model

Fig. 2 - Typical wellhead pressure behavior of waterflood injectors (South Belridge Diatomite)

Fig. 3 - Typical injection rate behavior of waterflood injectors (South Belridge Diatomite)

Fig. 4 - Performance of neural network model for training data set number 1.
Fig. 10 - Injection rate of test injector

Fig. 11 - Neural network prediction and suggestion for injection rate

Fig. 12 - Comparison between neural network prediction and suggestion with actual injector response
Fig. 13-Plan view of Phase II steam drive pilot.

Fig. 14-Typical injection rate behavior of IN2L steam injector and the performance of the neural network model.

Fig. 15-Typical wellhead pressure behavior IN2L steam injector and the performance of the neural network model.

Fig. 16-Performance of the network model for predicting the cumulative amount of steam injected for different injection pressure policy.