SPE 121394

Forecasting the Post-Fracturing Response of Oil Wells in a Tight Reservoir

Jhiin Joo, SPE, Dongrui Wu, SPE, Jerry M. Mendel, Alejandro Bugacov, SPE, University of Southern California

This paper was prepared for presentation at the 2009 SPE Western Regional Meeting held in San Jose, California, USA, 24–26 March 2009.

Abstract

This paper proposes a two-stage multi-system architecture for forecasting post-fracturing responses in a tight oil reservoir using historical fracturing data. The first stage predicts the 180-day cumulative liquid (oil + water) production directly, and the second stage uses differential correction to predict the prediction error resulting from the first stage. The final prediction is a combination of the two stages. 5-fold cross-validation is used in each stage, resulting in five forecasters for each stage. The average of the five predictions is taken as the output of the corresponding stage. Each of the five forecasters in each stage consists of three independent subsystems (Location, Completion and Fracturing), whose inputs are subsets of the well properties. The Location subsystem is constructed by a weighted average, whereas Completion and Fracturing are constructed by fuzzy logic systems. The parameters of the three subsystems are optimized simultaneously using simulated annealing. The final design achieved over 70% prediction accuracy for more than 96% of the testing wells. The main advantages of our approach are that 1) it does not require a large training dataset; 2) it can cope well with incomplete data entries and uncertainties; and, 3) the redundancy in the input parameters is used to improve accuracy.

1. Introduction

This paper focuses on the prediction of post-fracturing productions in a tight oil reservoir where all producing wells need to be fractured due to extremely small pore size, high porosity and low permeability of the formation [1]. Previous efforts were made on classifying the post-fracturing productions into several levels, but no simple relationships were found between fracturing parameters and post-fracturing productions. Additionally, the application of fracturing simulators did not prove beneficial for this particular reservoir.

In this paper, we view this problem as forecasting (prediction) instead of classification and focus on the prediction of the180-day cumulative liquid production (oil + water) after fracturing. Our database consisted of completion and fracturing data for 344 wells fractured by two different vendors during a period of eight years. A complete entry in the database consists of about 30 different parameters, as shown in Fig. 1. These parameters were partitioned into three groups[1], e.g., Location, Completion and Fracturing. Some of these inputs are hierarchically organized, e.g., each zone in Completion includes a group of "local" parameters (feet of perforations, number of holes, etc), and each stage in Fracturing also includes a group of "local" parameters (slurry/sand/pad volumes, pressure, and treatment rate).

The challenges of this project were:

- The database is small compared with the number of input; hence, the forecaster needs to be designed in such a way that it can be trained extensively using a small database.
- There are missing values in the database; hence, the forecaster needs to be able to cope well with incomplete data entries.
- There are errors in the database, and it is very difficult to identify them; hence, the forecaster needs to be able to cope well with uncertainties.

In this paper a two-stage multi-system architecture is proposed, as shown in Fig. 2. The first stage predicts the 180-day[2] cumulative liquid (oil + water) production directly, and the second stage uses differential correction to predict the prediction error resulting from the first stage. A multi-system approach is used because then each sub-system has only a few free parameters, which can be tuned using a small training dataset. Because the methods to construct the two stages are the same, except that their predictions are different, only the method to construct the first stage is described in detail next.

[1] Our partition is similar to the one used in [2].

[2] The 180-day cumulative liquid (oil + water) production is used in this paper; however, our approach can be easily extended to the forecast of any other measure of productivity.

2. Design of the first stage

5-fold cross-validation [5] (Fig. 3) was used to partition our database into five folds for training and testing, leading to five individual forecasters. There were three sub-systems (Location, Completion and Fracturing) for each forecaster, which can be constructed independently using different methods. The parameters of the three sub-systems in each forecaster were optimized simultaneously using Simulated Annealing [6]. The output of the first stage, which is a rough prediction of the 180-day liquid production, was computed as the average of the five forecasters.

2.1. Data pre-processing

As has been mentioned in the Introduction, there are missing values and inaccurate data in the database; so, the database must be conditioned before it can be used.

It was found that some wells were out of production (downtime) for a certain amount of time during the 180-day cumulative period, e.g., Well 1 may actually produce 172 days in the 180-day period, whereas Well 2 may produce only 164 days. To reduce this kind of uncertainties, the cumulative production values in the original database were compensated using downtime. Denote the actual production dates of Well *i* in the 180-day period as m_i and the uncompensated 180-day cumulative liquid production in the database as y_i. Then, the downtime-compensated 180-day cumulative liquid production is computed as

$$y_{i,c} = 180 y_i / m_i \quad (1)$$

All inputs and outputs in the database also need to be validated to remove bad data, outliers, etc. Generally, the validation process consists of three steps:

1) *Bad data processing* to remove unreasonable data:
   - For production, the following are considered as bad data: a) Any of 30-, 60-, 90- or 180-day production that is equal to or less than 0; and, b) The 60-day cumulative liquid production is smaller than or equal to the 30-day cumulative liquid production, the 90-day production is smaller than or equal to the 60-day production, etc.
   - There is no bad data processing for the Location subsystem.
   - For the Completion subsystem, bad data mean wells with incomplete completion data and wells with zero feet of perforations and/or zero number of holes.
   - For the Fracturing subsystem, bad data mean wells with incomplete slurry/sand/pad volumes.

2) *Outlier processing* using a Box and Whisker test [4]. Outliers are points that are unusually too large or too small. A Box and Whisker test is usually stated in terms of first and third quartiles and an inter-quartile range. The first and third quartiles, $Q(0.25)$ and $Q(0.75)$, contain 25% and 75% of the data, respectively. The inter-quartile range, *IQR*, is the difference between the third and first quartiles; hence, *IQR* contains 50% of the data between the first and third quartiles. Any datum that is more than $1.5IQR$ above the third quartile or more than $1.5IQR$ below the first quartile is considered an outlier.

3) *Tolerance limit processing* [4]. If a datum lies in $[m - k\sigma, m + k\sigma]$, then it is considered good; otherwise, it is rejected. *k* is determined such that we have 95% confidence that the given limits contain at least 95% of the available data. Note that depending on the number of remaining wells after Step (2), *k* in this step may be different for each parameter.

The production data were validated first. Wells with invalid production data were removed from further consideration because if the production data of a well were not valid then that well cannot be used by any subsystem. Next, the inputs to Location, Completion and Fracturing subsystems were validated independently, and for each subsystem the wells with valid production data were partitioned into two groups:

- *Valid wells*, which have valid inputs for the current subsystem (but may not be valid for the other two subsystems); and,
- *Invalid wells*, which do not have valid inputs for the current subsystem (but may be valid for another subsystem). These wells will be removed from further consideration in that subsystem.

Once the three valid well sets for the three subsystems were obtained, they were further partitioned into two groups:

- *Common wells*, which are the common set of wells from the three valid well sets. The common wells were used in both training and testing of the five forecasters.
- *Extra wells*, which are the remaining wells in the valid well sets for Location after removing the common wells. Generally, extra wells for the three subsystems were different. They were used in training of the Location subsystem, but not in testing.

For example, if valid wells for Location are {1, 2, 3, 5}, valid wells for Completion are {1, 2, 3, 4}, and valid wells for Fracturing are {2, 3, 5, 6}, then the common wells are {2, 3}, and extra wells for Location are {1, 5}.

2.2. Multi-system architecture

Each forecaster consists of three subsystems, as shown in Fig. 4. Input parameters for the Location subsystem are the X and Y coordinates of a well. The Completion subsystem uses total feet of perforations in each zone, total number of holes in each zone, permeability in each zone and the total number of stages. The inputs to the Fracturing subsystem are the total Pad, Sand and Slurry volumes for each well. Depending on the input type, different subsystems may use different computational methods (e.g., fuzzy logic systems (FLSs), artificial

neural networks, linear regression, etc) to model its behavior. Note that in our approach each subsystem is constructed independent of each other without considering the methods and outputs of other subsystems. Once the three subsystems are constructed, their outputs are combined using a linear (arithmetic) or nonlinear (interval weighted) average.

2.2.1. The Location Subsystem

The Location subsystem has two inputs: X and Y coordinates of each well. There are several different methods that can be use to represent the coordinate information, e.g., kriging, inverse distance weighting (IDW) [8], nearest neighbors, etc. The basic ideas of these methods are similar. In this paper the IDW method was used for its simplicity.

The generic equation for IDW is

$$y_L^k = \frac{\sum_{i=1}^{n(k)} y_i(k) w_i(k)}{\sum_{i=1}^{n(k)} w_i(k)} \tag{2}$$

where y_L^k is the estimated liquid production (from the Location subsystem only) of the k^{th} well, $y_i(k)$ is the production of its i^{th} neighbor, which is included in the training dataset, and $w_i(k)$ is the weight for that neighbor. In our approach $n(k)$ is the number of neighbors within a certain radius, *r*. Note that $n(k)$ may vary from well to well. If all neighbors have distances larger than *r*, then y_L^k is set equal to the production of the nearest neighbor, no matter how far away it is.

The general strategy for constructing $w_i(k)$ is to assign larger weights to nearby neighbors and smaller weights to far away neighbors. Particularly, in this paper

$$w_i(k) = \left(\frac{d_i(k)}{\min_{\forall j} d_j(k)} \right)^{-\beta} \tag{3}$$

where $d_i(k)$ is the distance between the i^{th} well and the k^{th} well, and β is a positive constant. Observe that there are only two parameters, β and *r*, to be optimized in the Location subsystem.

2.2.2. The Completion Subsystem

Three parameters, the total feet of perforations, the total number of holes, and the number of stages, were used in the Completion subsystem; however, they were not used directly. Because the perforations were performed in different zones, whereas each zone has different permeability (porosity) and hence different influence to the post-fracturing response, the zonal permeabilities were used to weight the inputs. The actual inputs to the Completion subsystem were the *permeability weighted total feet of perforations (*f*),* the *permeability weighted total number of holes* (h), and *the total stage number* (h), i.e.[3],

$$f = \sum_{i=1}^{13} p_i f_i \tag{4}$$

$$h = \sum_{i=1}^{13} p_i h_i \tag{5}$$

$$s = \sum_{i=1}^{13} s_i \tag{6}$$

where 13 is the total number of zones, f_i is the feet of perforations in Zone *i*, h_i is the number of holes in Zone *i*, and p_i is the permeability of Zone *i*. Note that the total stage number is not weighted.

The Completion subsystem was implemented by a FLS with rules in the form of:

R: *IF* f *is* F, h *is* H, *and* s *is* S, *THEN* y_c *is* y.

where F, H and S are fuzzy sets, and *y* is a crisp number. Four[4] Gaussian membership functions (MFs) were used for each of *f* and *h*, and one *s*-shaped MF was used for *s*. Consequently, there were a total of 16 rules. Because each Gaussian MF and *s*-shaped MF is completely determined by two parameters, there were (2×4) + (2×4) + (2×1) = 18 parameters for the MFs and 16 parameters for the consequents, i.e., a total of 34 parameters need to be optimized for the Completion subsystem. Note that the *f*, *h* and *s* domains may be partitioned more finely to achieve better performance; however, the number of parameters to be tuned increases rapidly.

2.2.3. The Fracturing Subsystem

Each fracturing job was completed in several stages. The total pad, slurry and sand volumes from all stages were used as inputs to the Fracturing subsystem. A FLS was also used to construct the fracturing subsystem, and its rules were in the form of:

R: *IF pad is* P *and slurry is* Sl *and sand is* Sa, *THEN* y_F *is* y.

where *P*, *Sl* and *Sa* are fuzzy sets, and *y* is a crisp number. Three Gaussian MFs were used for the total sand volume, and two Gaussian MFs were used for the total slurry volume and total pad volume. Consequently, there were a total of 2×3+2×2×2+3×2×2 = 26 parameters to be optimized.

2.3 Optimization

The outputs of the three subsystems can be combined using different methods. Two methods are considered in this paper: a linear arithmetic average (AA) and a non-linear method called interval weighted average (IWA) [7].

[3] *f*, *h* and *s* can be normalized to the [0, 1] interval so that linguistic descriptions can be assigned to their membership functions, e.g., small, large, etc; however, whether or not *f*, *h* and *s* are normalized will not change the prediction performance, whereas the normalization takes extra time. So, they are not normalized in this paper.

[4] The numbers of MFs for Completion and Fracturing subsystems were obtained by trial and error.

These weights, as well as the parameters of the three subsystems, were optimized simultaneously using Simulated Annealing [6].

The objective function of simulated annealing was the sum of the squared forecasting errors. Let y_i be the actual 180-day liquid production of Well i, and y_i^p be the prediction. Then, the objective function is

$$J = \sum_{i=1}^{N}(y_i - y_i^p)^2 \quad (7)$$

where N is the number of training examples. The change of J with respect to the number of iterations is shown in Fig. 5. It converges as the number of iterations increases.

3. Results

The prediction performance was evaluated based on the number of wells whose forecasted 180-days liquid production is outside the ±30% and ±20% error bound and the mean-squared errors. Figs. 6 and 7 show the performance of the proposed approach for wells fractured by two different service companies, Vendor A and Vendor B, respectively. These companies used different fracturing methodologies and cannot be predicted using a single forecaster.

The top-left figures in Figs. 6 and 7 show the performances of baseline linear regression models. Their inputs was a 8-dimensional vector, which is a combination of the inputs to the three subsystems, i.e., x and y coordinates, permeability weighted total feet of perforations, permeability weighted total number of holes, total stage number, total pad/slurry/sand volumes, and the output is the average of the five forecasters. Observe that many predictions were out of the 30% error bound. This is not surprising because it does not seem reasonable to model a complex fracturing process using a simple linear regression model.

The top-right figures in Figs. 6 and 7 show the performance of the multi-system approach, where the three subsystems were combined using an arithmetic average (AA) without differential correction (DC). Observe that this approach dramatically outperformed the linear regression model; however, the mean squared error was still large.

The bottom-left figures of Fig. 6 and 7 show the results when AA was used to combine the three subsystems and DC was also implemented. Observe that this approach outperformed the two previous methods in terms of both the number of wells outside accuracy bounds and the mean squared errors.

Finally, the bottom-right figures of Fig. 6 and 7 show the results when the interval weighted average was used to combine the three subsystems and DC was also implemented. Observe that it gave the best performance among the four approaches, i.e., both the number of wells outside the accuracy bounds and the mean squared errors were the smallest. This approach can not only follow the trend of values, something which linear regression cannot even do, but also give predictions within the 30% error band for more than 96% of the wells in our database and within the 20% error band for over 80% of the wells.

In summary, the main advantages of the proposed approach include:

- *Flexibility*. According to the form of the inputs, each subsystem can be constructed in its own way, and different subsystems may use different processing methods, e.g. fuzzy logic systems, artificial neural networks, decision trees, linear regressions, etc.
- *Utilization of the redundancy in the inputs to improve performance.* By redundancy it meant that some inputs may be related to other inputs, e.g., the injected volumes should be partially dependent on the feet of perforations and number of holes in each zone. The sub-systems approach reduces the redundancy in each sub-system, but the full system does utilize the redundancy in the inputs. Moreover, this approach reduces the "curse of dimensionality."
- *Easiness to incorporate expert rules.* Because each subsystem has only a few inputs, people can understand it and give guidelines for its rules.
- *Easiness to handle missing values.* If a single full system is constructed using all inputs, a missing value in a data entry usually means that data entry cannot be used, or the missing value must be estimated. On the other hand, in our approach a missing value only affects one subsystem.

4. Conclusion

A two-stage multi-system architecture has been proposed in this paper to predict the 180-day cumulative liquid productions in a tight oil reservoir. This data-driven forecasting system achieved satisfactory prediction accuracy given the relatively small training data size and the complexity of the problem. The forecaster constructed in this paper is part of our ongoing work on post-fracturing response optimization where the forecaster is used to estimate the performance of a given fracturing design during the optimization process. The fracturing optimization part of our work will be reported in a future publication.

Acknowledgments

This study was funded by the Center of Excellence for Research and Academic Training on Interactive Smart Oilfield Technologies (CiSoft). CiSoft is a joint University of Southern California-Chevron initiative. The authors would like to thank Prof. Iraj Ershaghi and Angela M. Madding for their valuable discussions.

References

[1] K.A. Hejl, A.M. Madding, M.F. Morea, C.W. Glatz, J. Luna, W.A. Minner, T. Singh and G.R. Stanley, "Extreme Multistage Fracturing Improves Vertical Coverage and Well Performance in the Lost Hills Field," *SPE 101840*, SPE Annual Technical Conference and Exhibition, San Antonio, TX, September 2006.

[2] S. Mohaghegh, S. Reeves and D. Hill, "Development of an intelligent systems approach for restimulation candidate selection," *SPE 59767*, SPE/CERI Gas Technology

Symposium, Calgary, Alberta, Canada, April 2000.

[3] Dengen Zhou, Jairam Kamath, Francois Friedmann, and Mike Morea, "Identifying Key Recovery Mechanisms in a Diatomite Waterflood," *SPE 75142*, SPE/DOE Thirteenth Symposium on Improved Oil Recovery, Tulsa, OK, April 2002.

[4] R. W. Walpole, R. H. Myers, A.L. Myers and K. Ye, *Probability & Statistics for and Scientists*," 8th Edition, Prentice-Hall, Upper Saddleback River, NJ, 2007.

[5] Kohavi, Ron, "A study of cross-validation and bootstrap for accuracy estimation and model selection," International Joint Conference on Artificial Intelligence (IJCAI): 1137–1143, Morgan Kaufmann, San Mateo, 1995.

[6] S. Kirkpatrick, C. D. Gelatt and M. P. Vecchi., "Optimization by simulated annealing," Science, Vol. 220, pp. 671-680, 1983.

[7] F. Liu and J. M. Mendel, "Aggregation Using the Fuzzy Weighted Average as Computed by the Karnik-Mendel Algorithms, "IEEE Trans. on Fuzzy Systems, Vol. 16, No. 1, pp.1-12, 2008.

[8] Patrick M. Bartier and C. Peter Keller, "Multivariate interpolation to incorporate thematic surface data using inverse distance weighting (IDW)," *Computers & Geosciences*, vol. 22, no. 7, pp. 795-799, 1996.

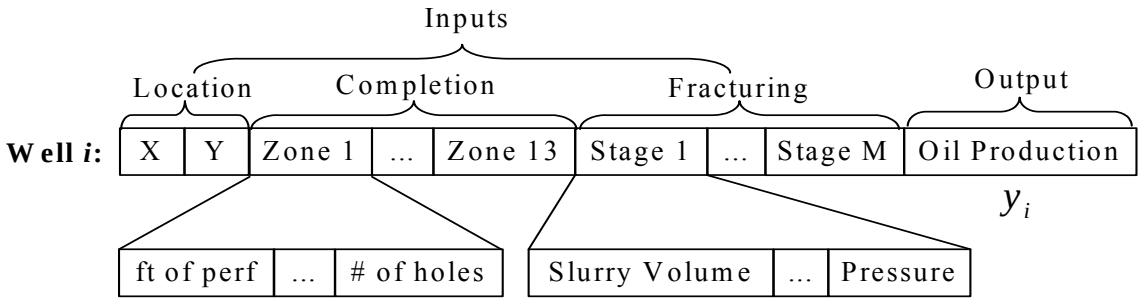

Fig. 1. A complete data entry in the database.

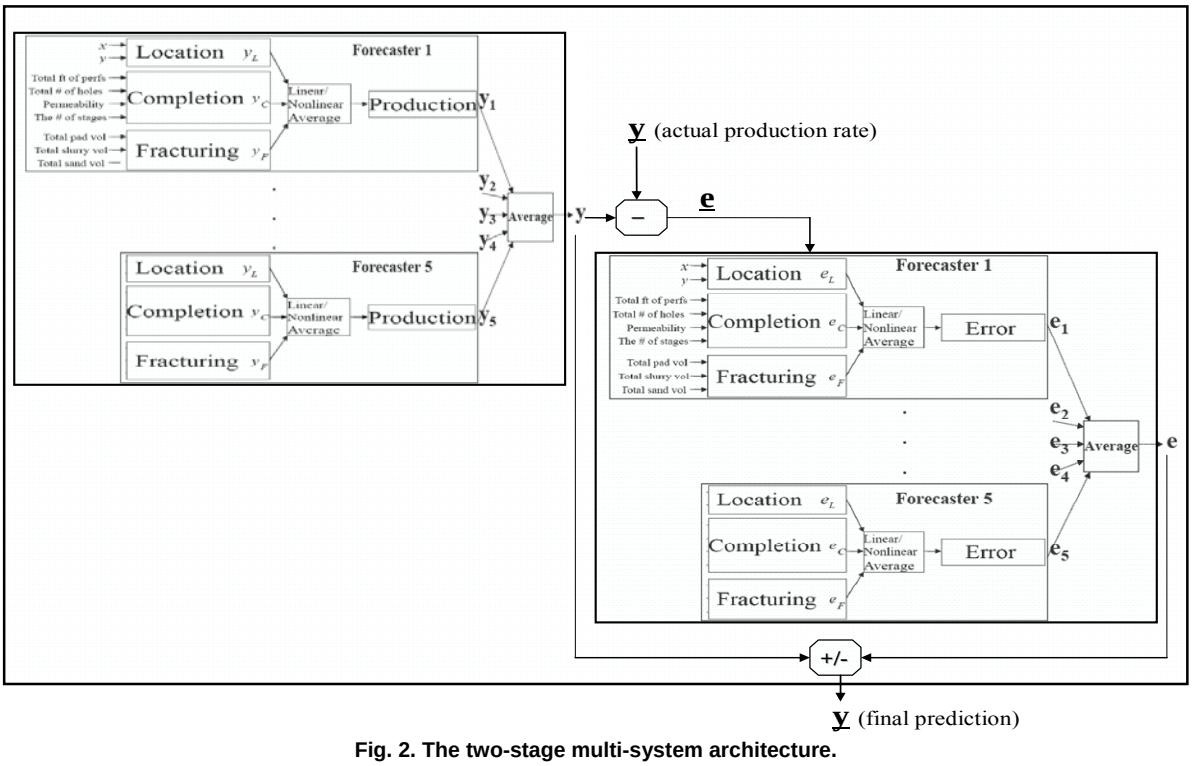

Fig. 2. The two-stage multi-system architecture.

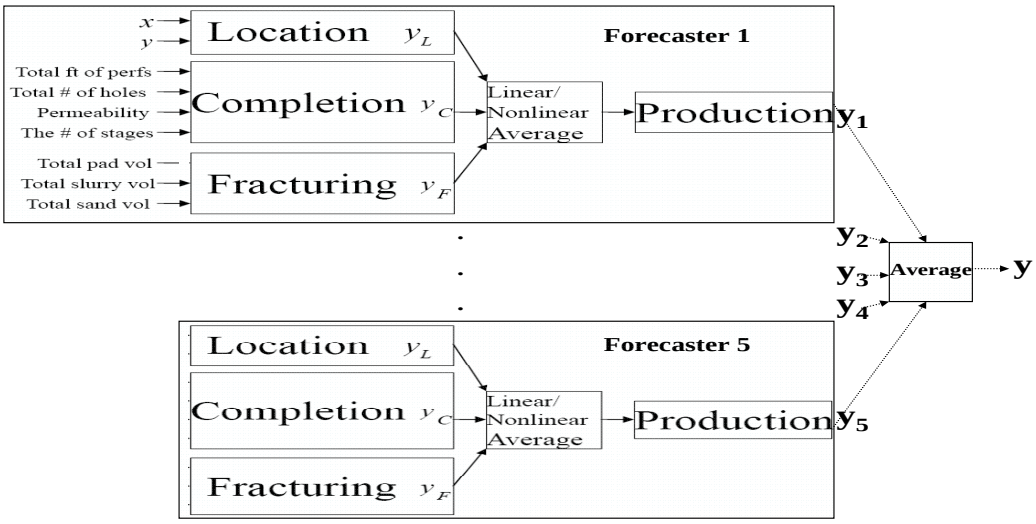

Fig. 3. The 5-fold cross-validation approach.

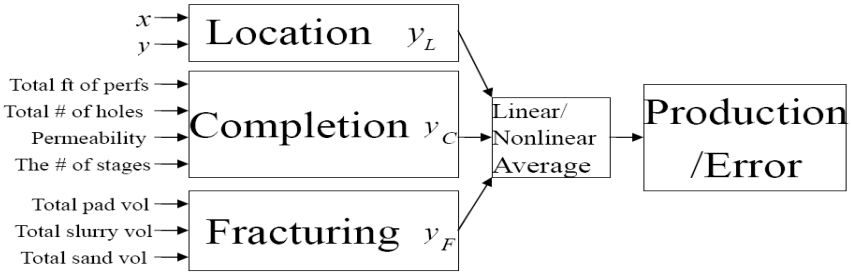

Fig. 4. Inputs and Output parameters for the three subsystems.

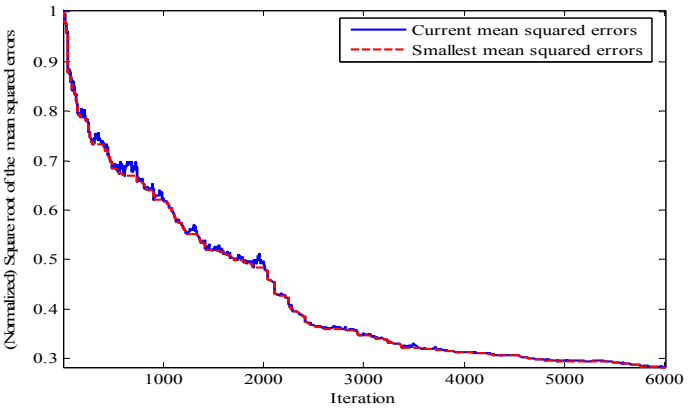

Fig. 5. Performance of Simulated Annealing used in optimizing.

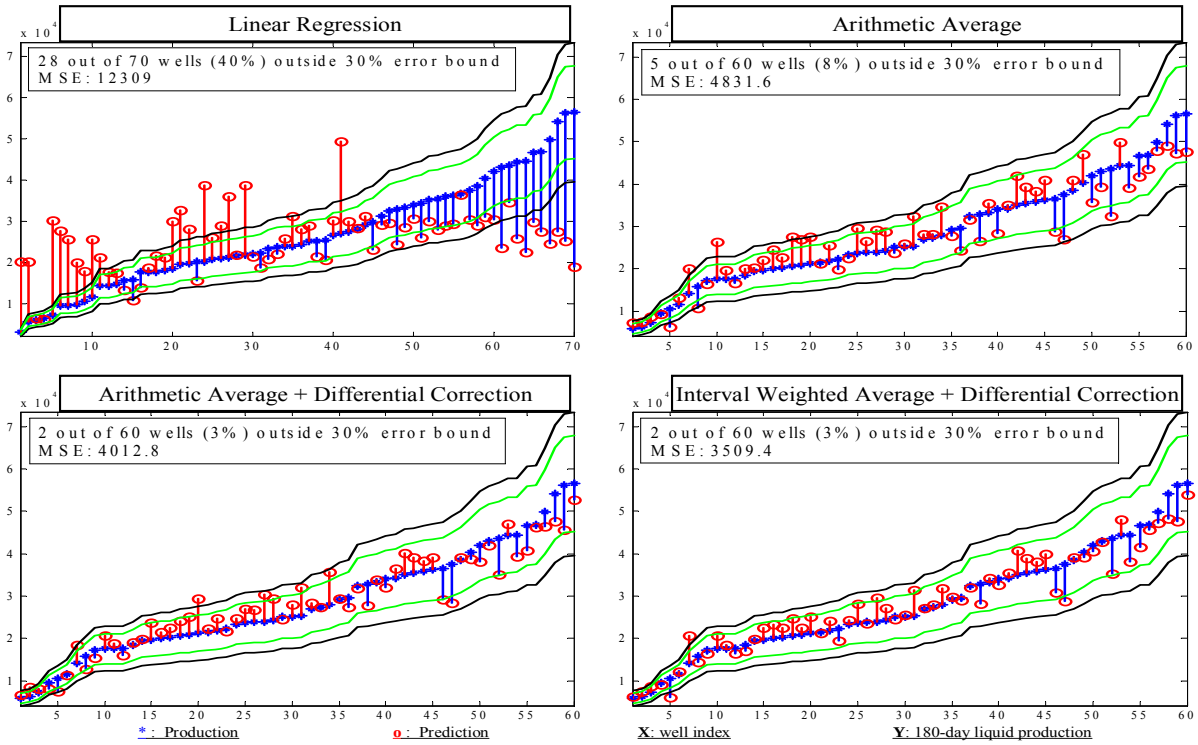

Fig. 6. Forecasting of 180-day cumulative liquid production for wells fractured by Vendor A.

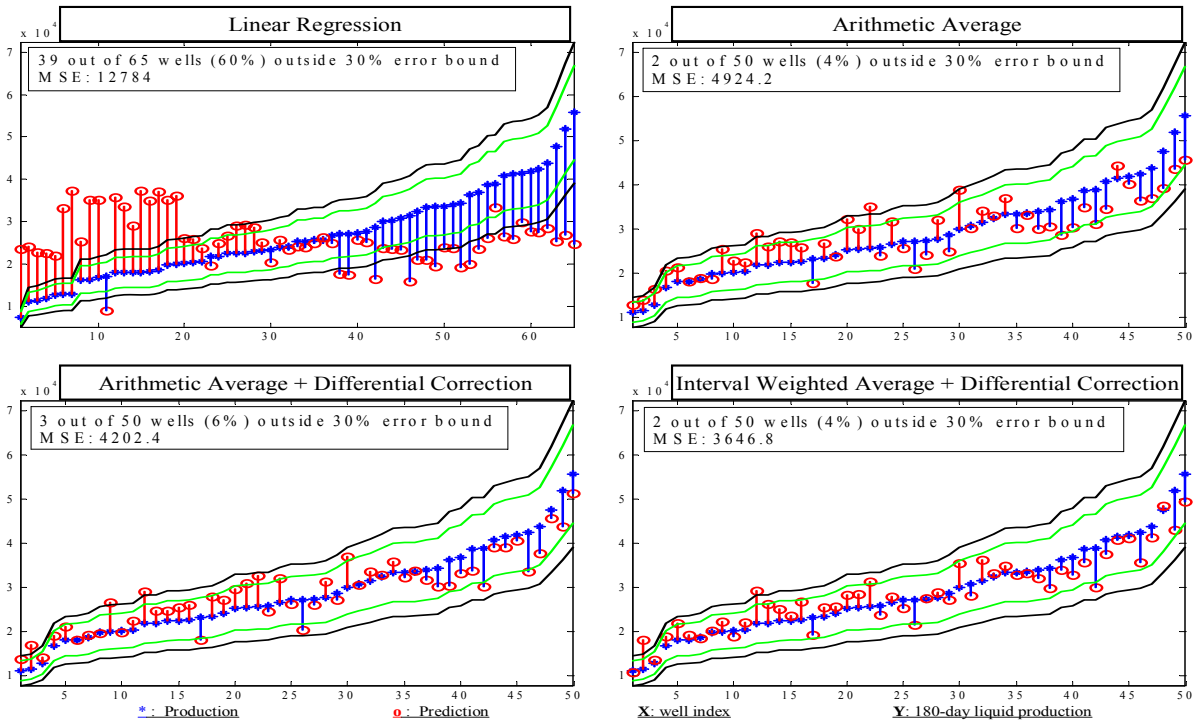

Fig. 7. Forecasting of 180-day cumulative liquid production for wells fractured by Vendor B.