An approach for noise reduction in inverse problems, applied to characterization of oil reservoirs with well-test data

C. Minutti, S. Gomez, G. Ramos
National University of Mexico, Institute of Applied Mathematics, Mexico City, Mexico
E-mail: carlos.minutti@iimas.unam.mx

Abstract. Reservoir characterization is an inverse problem where parameter values of the porous media are estimated. In this problem, the pressure and its log-derivative curves are used in the fitting process. However, the numerical differentiation to generate the log-derivative curve, is an ill-posed problem where the noise in the data can be largely propagated. This noise can produce several spurious local minima of the objective function and prevents to get a precise approximation to the parameters. Therefore, an efficient noise reduction method is needed to achieve enough accuracy to reproduce the data.

In this work, we explore three noise reduction methods applied to well-test data. These methods are based on multi-step finite differences and splines, but using a machine-learning approach.

In addition, a new method is proposed which mixes multiple data curves of the model, in an optimal curve. We show how these methods which use information from the mathematical model are more efficient for noise reduction. The proposed methods can be used in many inverse problems where there is a mathematical model that describes the measured data.

1. Introduction

In many inverse problems, where the parameters of a mathematical model are identified through measured data, the noise in the data is a major problem, especially in complex models (with a large number of parameters).

The characterization of oil reservoirs through well tests is one of these problems. In well test analysis the log-derivative curve (the derivative of the measured pressure with respect to the logarithm of time), is often used in the objective function during the optimization process, as it indicates several characteristics of the reservoir. However, since the numerical derivation is an ill-posed problem, the noise of the data is propagated making it difficult to achieve a good estimation of the parameters of the model.

For the reservoir characterization, we use the Triple Porosity-Double Permeability (TP-DP) model by Camacho [2], in which it is necessary to identify up to 10 parameters for a real case (see [5], [6] and [8] where reservoir characterization using the TP-DP model, is described). The solution of the model for a given set of parameters, produces the curve of the pressure in the well under study. Due to the number of parameters to be identified, this model can give more information of the porous media than the conventional double-porosity model of Warren and Root [11]. Thus, it is necessary a noise reduction method to be able to produce the so called
type-curves of the pressure and the log-derivative, that have to be fitted to produce accurate estimate of the parameters of the model.

The Bourdet algorithm [1] based in a finite difference approach, has been the standard method to calculate the log-derivative curve of well-test data, but it has shown to be inefficient when the noise level is high.

Lane et al. [7] address the problem of noise reduction in well-test data for a simpler model, using splines in which the smoothing parameters are optimized. Later, Escobar et al. [3] also address the problem and compare different noise reduction methods, which are based on finite difference schemes (the Bourdet algorithm with different values of $L$ and the derivative of interpolated polynomials) and splines, concluding that splines is the most effective method. However, the finite differences based on the interpolation of polynomials at several points, considered by Escobar et al. increase the noise in the data.

The approach of using multi-step finite differences and/or splines, to reduce the noise in the data, is well extended in many different areas.

In this study, we contrast three methods of noise reduction: Multi-step finite differences, Splines and a Mixture of sub-optimal curves.

In both, splines and Multi-step finite differences, we add a variation where information from the model (the TP-DP model) itself is used to determine the optimal parameters of each of the methods. Being splines a method widely used in different areas to reduce noise in the data, we also study the effect of another type of splines: regression splines.

In the first part of the paper we present the methodology of how the classic algorithms of multi-step finite differences and splines are adapted to a machine-learning approach and the proposed method of mixture of sub-optimal curves is described. In the second part, we present the results obtained when applying these noise reduction methods to a large data set, measuring the error of the method and also its effect in the parameter estimation error of the inverse problem.

2. Methodology

Considering a machine-learning approach, we use the TP-DP model to generate a first group of 1,000 random synthetic data sets, for which Gaussian noise was subsequently added. In each of the methods to be studied, this data set is used to train the algorithms, that is, to determine the optimal value of the parameters that minimize the error between the noisy and noise-free data. Later a second group of 1,000 random synthetic data sets is generated, for which noise is also added. This data set is the cross-validation set and we use this data to compare the performance of the proposed algorithms, with the classic approach and with the optimized parameters. The purpose of determining the optimal parameters of each algorithm with a training data set, is to make the method to respond better to the kind of data that produces the mathematical model of the inverse problem.

The error due to the noise reduction algorithm at each data point is measured by:

$$ Error = \left| \frac{d'_{\text{algorithm}} - d'_{\text{noise-free}}}{d'_{\text{noise-free}}} \right| $$

where $d'_{\text{noise-free}}$ is the log-derivative of the data without noise, and $d'_{\text{algorithm}}$ is the log-derivative of the obtained data, after applying the noise reduction algorithm to the noisy data.

However, to compare the performance of each method we use the mean and median value of $\log_{10}(Error)$. This is because, since we are measuring a relative error, we are more interested in the order of magnitude of the error. Also because the mean is more sensitive to the existence of outliers than the median. When relative errors are used, $\log_{10}(Error)$ has a more symmetric distribution, so it is less affected by outliers.
2.1. Multi-step finite differences
In the case of noisy data, it does not make sense to use finite difference schemes that are based on interpolation of polynomials (as it is the case of the schemes presented by Fornber [4]), since they increase the noise of the numerical derivative, which is what happens also in Escobar et al. Instead, we use the schemes presented by Savitzky and Golay [10], where the base polynomial, fits the data in the least squares sense, following an idea similar to the Lanczos derivative. From this formula, we choose a scheme of 9 points for the first method:

\[ f'_0 \approx \frac{(f_1 - f_{-1}) + 2(f_2 - f_{-2}) + 3(f_3 - f_{-3}) + 4(f_4 - f_{-4})}{60h} \]  

(2)

For the second method, we use the scheme:

\[ f'_0 \approx \frac{1}{4} \left( \frac{\omega_1 (f_1 - f_{-1})}{2h} + \frac{\omega_2 (f_2 - f_{-2})}{4h} + \frac{\omega_3 (f_3 - f_{-3})}{6h} + \frac{\omega_4 (f_4 - f_{-4})}{8h} \right) \]  

(3)

where we determine the weights \( \omega_i \) by minimizing the error obtained with the training data set. Subsequently, the errors of both methods are compared using the cross-validation data set.

2.2. Splines
For splines, we consider three cases. First, a classic approach where a cubic spline is used, and the smoothing parameter is determined by cross-validation, that is, for each one of the 1,000 data sets, a part of the data is used to determine the spline coefficients, and the other part to determine the smoothing parameter, selecting in each case the value that minimizes the quadratic error.

In the second approach, we use optimization to determine a single smoothing parameter value, which minimizes the error of the entire training data set and the same value will be used in the cross-validation data set.

In the third approach we use regression splines (see [9]), where unlike the cubic splines, it is possible to use other types of base functions to approximate the data. In order to have a good result, the base functions must be similar or related to the functions that are generated by the mathematical model which is used to fit the measured data. In the case of the TP-DP model, the asymptotic approximations of the model at early and late times (which appear in Camacho et al. [2]), are used as base functions. The smoothing parameter is determined in the same way as in the previous case, using the training data set to optimize its value.

The performance of each algorithm is determined by measuring the error in the cross-validation data set.

2.3. Mixture of sub-optimal curves
We propose a method that we call mixture of sub-optimal curves, which consists of using the noisy (pressure) data to perform a series of quick optimizations to get the optimal parameters of the inverse problem model (in this case, the TP-DP model) from different starting points, in order to obtain multiple local optima or approximations to them. For this case, 40 sub-optimal curves were obtained from the TP-DP model for each of the 1,000 cases in the training data set. Each of the curves is combined with a weight that is determined by minimizing the quadratic error between the combined curve and the noisy data. A regularization parameter is added to the model in order to avoid problems of singular matrices or overfitting. Both, the best number of sub-optimal curves to be used and the value of the regularization parameter, are obtained using the training data set.

The vector of weights for each of the sub-optimal curves is determined by:

\[ b = (X'X + I\lambda)^{-1}X'y \]  

(4)
where $\lambda$ is the regularization parameter, $X$ is a matrix whose columns are the different sub-optimal curves values, $y$ is a vector which contains the noisy data set and $I$ the identity matrix. The noise-reduced curve and its log-derivative are obtained with $b'X$ and $b'X_d$, where $X_d$ is the matrix whose columns are the log-derivative of the columns of $X$. Once these parameters are determined, the process is applied to the cross-validation data set to determine the error of the method.

2.4. Estimation of the error of the model parameters

In order to measure the impact of these noise reduction methods in the estimation of the model parameters, we perform an optimization to obtain the parameters of the TP-DP model, whose solution should reproduce the synthetic data in the cross-validation data set, following the procedure described in Minutti et al. [8], but providing as a starting point to the optimization algorithm, the true value of the parameters with which the noise-free data was generated. Because the optimization algorithm has as starting point the true value of the parameters, any error in the parameters is due to the noise in the data.

To measure the error of each parameter we use the following formula:

$$err(\hat{\theta}) = \frac{|\theta - \hat{\theta}|}{\max_{\theta} \theta - \min_{\theta} \theta}$$

where $\theta$ is the true value of the parameter, $\hat{\theta}$ is the estimated value given by the optimization algorithm, and $\max_{\theta} \theta - \min_{\theta} \theta$ is the length of the interval of possible values that the parameter can take. Thus, equation (5) measure the absolute error expressed between 0 to 1.

3. Results

Using the cross-validation data set, the mean and median of $\log_{10}(Error)$ are determined for each of the methods described above, obtaining the values in Table 1.

<table>
<thead>
<tr>
<th>Multi-step finite differences</th>
<th>Smoothing Splines</th>
<th>Mixture of sub-optimal curves</th>
</tr>
</thead>
<tbody>
<tr>
<td>FD_9_CLS</td>
<td>SPL_CLS</td>
<td>MIX_OPT_C</td>
</tr>
<tr>
<td>FD_9</td>
<td>SPL</td>
<td></td>
</tr>
<tr>
<td>Mean</td>
<td>-1.6257</td>
<td>-1.5021</td>
</tr>
<tr>
<td>Median</td>
<td>-1.7406</td>
<td>-1.6342</td>
</tr>
</tbody>
</table>

It can be observed that, as in multi-step finite differences and also in splines, the methods where the parameters of the algorithms were determined by means of a training set (FD_9, SPL, REG_SPL), are more efficient than using a classic method (FD_9_CLS, SPL_CLS). It is also observed that the method of regression splines (REG_SPL) is more efficient than the other two methods of splines, and the method of mixture of sub-optimum curves is better than any other of the analyzed algorithms. Specifically, the mixture of sub-optimal curves (MIX_OPT_C) has only 53.6% of the noise of regression splines, the second best method ($10^{-2.5310} \approx 0.5363$), showing that the use of information of the model of the inverse problem, produces a more efficient method for noise reduction.
Figure 1. Boxplot of \( \log_{10}(\text{Error}) \) for the trained algorithms.

Figure 2. Histogram of \( \log_{10}(\text{Error}) \) for the trained algorithms.
In Figures 1 and 2 the boxplots and histograms are shown for the methods where the algorithms are trained: Multi-step finite differences (FD-9), cubic splines with optimized smoothing parameters (SPL), regression splines with optimized smoothing parameters (REG_SPL) and the mixture of sub-optimal curves (MIX_OPT_C).

In Figures 3-5 we present three examples taken from the cross-validation data set, where the pressure and its derivative (log-derivative) are plotted on a log-log scale with the result of the three best methods: trained splines (cubic and regression splines), and mixture of sub-optimal curves. The best results are obtained with the mixture of sub-optimal curves. Cubic splines and regression splines have problems of oscillations, especially in the valleys of the log-derivative curves.

![Derivative with Central Differences](image1)

![Noise reduction with Splines](image2)

![Noise reduction with Regression Splines](image3)

![Noise reduction with a Mixture of Sub-optimal Curves](image4)

**Figure 3.** Example 1: Noise reduction with the three best methods.

In Table 2 we show the mean estimation error of the parameters of the TP-DP model, when we solve the inverse problem using the noisy data in the cross-validation data set and also when we use the noise-reduced data given by the trained algorithms. As we provide the optimization algorithm with the true value of the parameters as a starting point, the estimation error is due to the noise in the data.

For each parameter can be observed a significant reduction in the estimation error compared with the noisy data. For parameters like $\omega$, $\lambda_{mf}$, $\lambda_{mv}$ and $\kappa$, the mean estimation error is one-third of the error of the noisy data when the mixture of sub-optimal curves was used.

Although regression splines was a better noise reduction method than cubic splines, there was not much difference in the mean estimation error of the parameters.
Figure 4. Example 2: Noise reduction with the three best methods.

Table 2. Mean estimation error of the model parameters with different noise reduction methods

<table>
<thead>
<tr>
<th>Method</th>
<th>$\omega_f$</th>
<th>$\omega_v$</th>
<th>$\lambda_{mf}$</th>
<th>$\lambda_{mv}$</th>
<th>$\lambda_{vf}$</th>
<th>$\kappa$</th>
<th>$s$</th>
<th>$C_D$</th>
</tr>
</thead>
<tbody>
<tr>
<td>NOISY DATA</td>
<td>0.1862</td>
<td>0.1844</td>
<td>0.1008</td>
<td>0.0966</td>
<td>0.1615</td>
<td>0.1567</td>
<td>0.0148</td>
<td>0.0011</td>
</tr>
<tr>
<td>SPL</td>
<td>0.0955</td>
<td>0.0962</td>
<td>0.0403</td>
<td>0.0355</td>
<td>0.0775</td>
<td>0.0670</td>
<td>0.0012</td>
<td>0.0003</td>
</tr>
<tr>
<td>REG_SPL</td>
<td>0.0923</td>
<td>0.0945</td>
<td>0.0395</td>
<td>0.0365</td>
<td>0.0800</td>
<td>0.0639</td>
<td>0.0011</td>
<td>0.0003</td>
</tr>
<tr>
<td>MIX_OPT_C</td>
<td>0.0666</td>
<td>0.0623</td>
<td>0.0360</td>
<td>0.0340</td>
<td>0.0719</td>
<td>0.0488</td>
<td>0.0005</td>
<td>0.0001</td>
</tr>
</tbody>
</table>

4. Conclusions
From these results, it can be seen that for each method, the incorporation of information from the model, by training the algorithms with a synthetic data set, the error in the parameter values was reduced. Especially when we use regression splines with base functions derived from the model, we obtain better results than when using a classic cubic spline.

The proposed method, mixture of sub-optimal curves, combines the idea of using functions from the model as a basis and the training of the algorithm in order to get a set of optimized parameters to the problem. This method was the most efficient for noise reduction, having only 53.6% of the noise that has regression splines, the studied second best method.

When the inverse problem was re-solved for the cross-validation data set, we observe the impact of the different noise reduction methods on the mean estimation error, being the mixture of sub-optimal curves also the best method, having for many parameters, up to one-third of the...
estimation error as compared with the error of the noisy data.

Because in many inverse problems there is a mathematical model which is used to approximate or reproduce a real phenomena, the methods presented in this work, can be adapted to a large kind of these problems, as the model can be used to train the algorithms to reduce the noise in the observed data, especially mixing different sub-optimal solutions of the model in order to get a noise-reduced (and model consistent) data set.
References


