ANAHESS, A NEW SECOND ORDER SUM OF EXPONENTIALS FIT ALGORITHM, COMPARED TO THE TIKHONOV REGULARIZATION APPROACH, WITH NMR APPLICATIONS

Åsmund Ukkelberg [1,*], Geir Humborstad Sørland[2], Eddy Walter Hansen[3], Hege C. Widerøe[4]

[1,3] The University of Oslo (UiO), Boks 1072 Blindern, N-0316 Oslo, Norway
[2] Anvendt Teknologi AS, Munkvollveien 56, N-7022 Trondheim, Norway
[4] Statoil Research Centre, N-7005 Trondheim, Norway

ABSTRACT

The problem of fitting a sum of exponentials to second order data is often called the second order (or two dimensional) inverse Laplace transform (2D-ILT). The present work describes ANAHESS, a new inverse Hessian algorithm which handles this ill-posed numerical problem. The algorithm uses numerical expressions for the gradient and the Hessian matrix, and has performed well on various artificial and experimental data sets. The new algorithm is presented in a Nuclear Magnetic Resonance (NMR) context.

Keywords: *Sum of exponentials, T_1–T_2 relaxation, Hessian, Bayesian information criterion, Earth mover's distance*

1. THE SECOND ORDER SUM OF EXPONENTIALS FIT AND NUCLEAR MAGNETIC RESONANCE

2D NMR techniques are Nuclear Magnetic Resonance measurement techniques which produce second order data matrices. Such techniques are a frequently used tool in the characterization of fluid filled porous media, and they have found a wide variety of applications, such as analysis of food products, rock core samples and hydrating cement pastes. In these NMR techniques, the fitting of a sum of exponentials to the measured response surface is a vital tool. Such a fitting is often called the second order Laplace transform (2D-ILT). The goal of 2D-ILT is to transform the data from a sum of exponentially decaying components to a set of parameters characterizing the exponential decay of each component. Depending on the experimental setup, these parameters have a physical interpretation such as spin-lattice relaxation time T_1, spin-spin relaxation time T_2 or diffusion coefficient D. Although such a transform is notoriously ill conditioned numerically [1], algorithms that attempt to perform this transform are in widespread use.

Hürlimann *et al*. [2] investigated the properties of dairy products using second order D-T_2 and T_1–T_2 NMR experiments. Vital to these experiments was a 2D-ILT algorithm described by Venkataramanan *et al*. [3] and by Song *et al*. [4]. In food analysis, it is often seen that first order NMR measurements are insufficient in distinguishing between constituents such as fat and water, as their contributions to the total signal overlap too much. If second order experiments are done, however, 2D-ILT calculations and a study of T_2/T_1 ratios or D/T_2 ratios can be used to quantify food constituents [5]. In a similar study, McDonald *et al*. [6] used second order NMR measurements and the 2D-ILT algorithm made by Venkataramanan *et al*. [3] to investigate chemical exchange in cement pastes. Clear evidence was found for chemical exchange of water protons between gel and capillary pores in the cement pastes.

Toft Pedersen *et al*. [7] have devised a method for curve resolution of NMR data which is based on the Direct Exponential Curve Resolution Algorithm (DECRA) proposed by Winding and Antalek [8]. This approach is based on numerical tools which are common in chemometrics. Toft Pedersen *et al*. used their algorithm to determine fat content in minced meat, and also pre slaughter stress in processed meat.

Arns *et al*. [9] used 2D-ILT software when discussing susceptibility effects in oil reservoir rock core samples, with the objective of developing an NMR response simulation tool for X-ray-CT images, including the simulation of T_1–T_2 relaxation. It was found that T_1 and T_2 values depend on bulk vs. surface relaxation, and therefore depend upon the pore structure within the rock core sample.

Sørland *et al*. [10] investigated ways to improve the performance of 2D-ILT on rock cores containing oil and water. They found that by running a series of experiments at varying gradient strength, the different molecule

mobilities of oil and water could be used to separate the NMR signals from oil and from water. Doing this separation prior to 2D-ILT, enhanced the ability of 2D-ILT to extract valuable information about the rock core sample.

The above applications, with the exception of the method by Toft Pedersen *et al*. [7] involve algorithms which fit the fractional intensities to a grid of equally spaced values of T_1 and T_2 (This applies to the case of T_1/T_2 experiments. This discussion could equally well be applied to D/T_2 experiments. In the following, explanations will be restricted to T_1/T_2 experiments in order to illustrate the concepts). The widely used algorithm implemented by Godefroy *et al*. [11], based on work by Venkataramanan *et al*. [3] is an example of such an approach. The grid of preset T_1 and T_2 values is an input to the algorithms, and may not reflect the optimal T_1 and T_2 values of the system investigated. Thus, without any prior knowledge of the optimal T_1 and T_2 values, those values may not be represented in the chosen grid. Instead, the algorithms provide an approximation to a continuous distribution in T_1 and T_2. In contrast, we propose a new algorithm called ANAHESS which searches for the T_1 and T_2 values that fit the dataset optimally, and provides a set of separate and distinct components which together constitute a model for the system investigated, instead of an estimate of a continuous distribution in T_1 and T_2.

2. PROBLEM FORMULATION

Assume an experiment which produces a second order matrix **R** having *NSA* (number of samples) rows and *NSE* (number of sensors) columns. Assume also that the measured response is the linear sum of *NCO* different components, each component decaying exponentially with both the row index and the column index. The row-wise exponential decay follows an axis g_i, i=1..*NSA*, contained in a vector **g** of size *NSA*, while the column-wise exponential decay follows an axis t_j, j=1..*NSE*, contained in a vector **t** of size *NSE*. Each component is assumed to exhibit a row-wise decay with a characteristic parameter $T_{1,p}$, p=1..*NCO*, while the component's column-wise decay is assumed to have a characteristic parameter $T_{2,p}$, p=1..*NCO*. The magnitude of each component is determined by a factor a_p, p=1..*NCO*, while any permanent signal offset present is assigned to the variable a_0. The spacing between subsequent element in **g** and **t** may or may not be constant. Any element $R_{i,j}$, i =1..*NSA*, j =1..*NSE* in **R** is then assumed to follow the function:

$$R_{i,j} = f(a_0, \mathbf{a}, \mathbf{T}_1, \mathbf{T}_2)_{i,j} + E_{i,j} \quad \text{with}$$
$$f(a_0, \mathbf{a}, \mathbf{T}_1, \mathbf{T}_2)_{i,j} = a_0 + \sum_{p}^{NCO} a_p e^{(-(1/T_{1,p})g_i - (1/T_{2,p})t_j)} \tag{1}$$

with a_p, $T_{1,p}$ and $T_{2,p}$ being the elements with the index p=1..*NCO* in the vectors **a**, $\mathbf{T}_1$ and $\mathbf{T}_2$, and subject to the following inequality constraints:

$$\begin{aligned} a_p &\geq 0, \quad p = 1..NCO \\ T_{1,p} &\geq 0, \quad p = 1..NCO \\ T_{2,p} &\geq 0, \quad p = 1..NCO \end{aligned} \tag{2}$$

a_0 is a baseline offset which may be positive or negative, and $E_{i,j}$ is the contribution from noise. The parameters a_p, $T_{1,p}$ and $T_{2,p}$ are thus characteristic properties of the component with index p out of all the *NCO* components. The objective is to perform a least squares fit of the function f to the data in the matrix **R**. This may be done by minimizing the function:

$$SS_{res} = F(a_0, \mathbf{a}, \mathbf{T}_1, \mathbf{T}_2) = \sum_{i}^{NSA} \sum_{j}^{NSE} \{f(a_0, \mathbf{a}, \mathbf{T}_1, \mathbf{T}_2) - R_{i,j}\}^2 = \sum_{i}^{NSA} \sum_{j}^{NSE} \{a_0 + \sum_{p}^{NCO} a_p e^{(-(1/T_{1,p})g_i - (1/T_{2,p})t_j)} - R_{i,j}\}^2 \tag{3}$$

again subject to the constraints given by the inequalities (2). This problem is ill-conditioned, with very minute changes in the data leading to widely different solutions.

3. AN INVERSE HESSIAN ALGORITHM

The minimization of the function given in eq. (3) can be reformulated as an unconstrained minimization by introducing a new set of variables a_0, $\boldsymbol{\alpha}$, $\boldsymbol{\beta}$ and $\boldsymbol{\gamma}$ so that:

$$\begin{aligned} a_p &= \alpha_p^2, \quad p = 1..NCO \\ 1/T_{1,p} &= \beta_p^2, \quad p = 1..NCO \\ 1/T_{2,p} &= \gamma_p^2, \quad p = 1..NCO \end{aligned} \tag{4}$$

For all real values of a_0, $\boldsymbol{\alpha}$, $\boldsymbol{\beta}$ and $\boldsymbol{\gamma}$, minimizing $F(a_0, \boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\gamma})$ will yield a solution within the permitted region of $\mathbf{a}$, $\mathbf{T}_1$ and $\mathbf{T}_2$ which is defined by (2). If the variables a_0, $\boldsymbol{\alpha}$, $\boldsymbol{\beta}$ and $\boldsymbol{\gamma}$ are combined into one vector of variables $\mathbf{x}$ (of size $3NCO + 1$) as follows:

$$\mathbf{x} = \begin{bmatrix} a_0 \\ \boldsymbol{\alpha} \\ \boldsymbol{\beta} \\ \boldsymbol{\gamma} \end{bmatrix} \tag{5}$$

then an inverse Hessian (or Gauss-Newton) algorithm will be based on the equation:

$$\mathbf{x}_{k+1} = \mathbf{x}_k - \phi \mathbf{H}(\mathbf{x}_k)^{-1} \nabla(\mathbf{x}_k) \tag{6}$$

in which k is the index of the present iteration, $\mathbf{x}_k$ is the present position in the variable space, ϕ is the optimal scalar step length, $\mathbf{H}(\mathbf{x}_k)$ is the Hessian matrix of second derivatives of F at $\mathbf{x}_k$, and $\nabla(\mathbf{x}_k)$ is the gradient of F at $\mathbf{x}_k$ (Note that the gradient in this context is the steepest ascent of a mathematical function, not the magnetic field gradient!). ϕ is found using a suitable univariate minimization algorithm. If the search direction of $-\mathbf{H}(\mathbf{x}_k)^{-1} \nabla(\mathbf{x}_k)$ is not a descending direction at the present location $\mathbf{x}_k$, a steepest descent search along $-\nabla(\mathbf{x}_k)$ is performed. See Press *et al*. [12] for a detailed description of inverse Hessian algorithms. Analytical expressions of $\nabla(\mathbf{x}_k)$ and $\mathbf{H}(\mathbf{x}_k)$ are given in an appendix.

3.1. The Univariate Minimization

Each iteration in an inverse Hessian algoritm requires an univariate minimization along the search direction of $-\mathbf{H}(\mathbf{x}_k)^{-1} \nabla(\mathbf{x}_k)$. In the present MATLAB [13] implementation of ANAHESS, this is done using the routine *fminbnd*, which is based on golden section search and parabolic interpolation [12]. In the present implementation, the optimal step length ϕ is always somewhere between a minimum of 10^{-20} and a maximum of 20. Also, if the solution strays outside of the interval defined by the minima and maxima of $\mathbf{g}$ and $\mathbf{t}$, F is given a very high value.

3.2. Choice of the Initial Estimate For One Specific Choice of *NCO*

Kaufmann [14] has developed a non-iterative algorithm for the fitting of a sum of exponentials to first order data. The goal of the first order sum of exponentials fit is to fit a sum of exponentials to a vector $\mathbf{r}$ so that any element r_i, $i = 1..N$ in $\mathbf{r}$ follows the function:

$$\begin{aligned} r_i &= f(a_0, \mathbf{a}, \mathbf{b})_i + e_i \quad \text{with} \\ f(a_0, \mathbf{a}, \mathbf{b})_i &= a_0 + \sum_p^{NCO} a_p e^{b_p v_i} \end{aligned} \tag{7}$$

in which v_i is the element having index $i=1..N$ in the vector $\mathbf{v}$. Applying Kaufmann's algorithm on each row in R provides an estimate of $\gamma_p^2 = -b_p$, $p=1..NCO$. Likewise, Kaufmann's algorithm applied on each column provides an estimate of $\beta_p^2 = -b_p$, $p=1..NCO$. Once the β_p^2 and γ_p^2 estimates are available, estimates of a_0 and a_p, $p=1..NCO$ may then be found using the method of linear least squares.

If a 2D-ILT calculation with *NCO* components is attempted, and the results from a calculation on the same data set with *NCO*-1 components are available, then the *NCO*-1 calculation results are compared to the initial estimates from Kaufmann's algorithm, and taken into account when the *NCO* calculation is started.

3.3. Choice of Number of Components

A good criterion for choosing the correct number of components *NCO* depends on the descriptive power of the model and the parsimony of the model. In this case, the descriptive power rests on the model's ability to minimize the sum of squared residuals, while the parsimony is the model's dependence on a low number of free parameters. When dealing with a demanding modelling problem, a model's descriptive power and the parsimony typically pull in opposing directions. Improving one property usually diminishes the other.

One criterion for choosing a model is the Bayesian information criterion (*BIC*), introduced by Schwartz [15]. Let n be the number of observed data points, let p be the number of free model parameters, and ss_{res} be the sum of squared residuals. Then if the residuals are normally distributed, *BIC* has the form:

$$BIC = n\ln(\frac{SS_{res}}{n}) + p\ln(n) \tag{8}$$

In this equation, a good model fit gives a low first term while few model parameters give a low second term. When comparing a set of models, the model with the minimal *BIC* value is selected. In the context of ANAHESS, n is *NSA* x *NSE*, SS_{res} is given in equation 3, and p is equal to $3NCO + 1$, since a_0, $\mathbf{a}$, $\mathbf{T}_1$ and $\mathbf{T}_2$ are the free parameters in the model. In ANAHESS, *BIC* is calculated for each successive value of *NCO*, and the model with the minimal *BIC* is chosen as the final model.

3.4. Summary of ANAHESS

The elements outlined above may now be combined into the following algorithm:

I. Acquire the necessary input data: **R**, **g** and **t**.
II. Choose a suitable maximum number of components NCO_{max}.
III. For *NCO*=1 to NCO_{max} do (component loop):

a) Find an initial estimate $\mathbf{x}_0$ using Kaufmann's algorithm along the rows and columns of **R**.
b) If *NCO*>1 then compare $\mathbf{x}_0$ to the results from the previous calculation with *NCO*-1 components, and use the results to improve upon $\mathbf{x}_0$ if possible.
While no convergence in F and $\mathbf{x}_k$ do (inverse Hessian loop):
(A) Calculate the various functions given by (9).
(B) Calculate $\nabla(\mathbf{x}_k)$ according to (10).
(C) Calculate $\mathbf{H}(\mathbf{x}_k)$ according to (11).
(D) Check **H** for positive definiteness. Make **H** positive definite if necessary.
(E) Do a univariate optimization of ϕ in equation (6), and move from $\mathbf{x}_k$ to $\mathbf{x}_{k+1}$.
(F) Check for convergence in F and $\mathbf{x}_k$.
c) End inverse Hessian loop.
d) Extract **a**, $\mathbf{T}_1$ and $\mathbf{T}_2$ according to (4) and (5), and store *NCO*, a_0, **a**, $\mathbf{T}_1$ and $\mathbf{T}_2$.
e) Calculate *BIC* for the given *NCO*.

IV. End component loop.

At the end, this algorithm has produced a set of 2D-ILT solutions with *NCO* ranging from 1 to NCO_{max}, with one solution for each *NCO* value. The solution with the minimal *BIC* is finally chosen as the optimal ANAHESS solution.

4. A COMPARISON OF EXPONENTIAL FIT ALGORITHMS

Godefroy *et al*. [11] have developed a Tikhonov regularization MATLAB routine called twodlaplaceinverse.m [16], which is based an a strategy given by Lawson and Hanson [17], in which a regularization function or smoothing function is added to the data before minimizing the squared sum of errors. A singular value decomposition is then used to reduce data size. Instead of computing a specific set of *NCO* components with accompanying **a**, $\mathbf{T}_1$ and $\mathbf{T}_2$ vectors, the results are computed as a second order distribution and presented as a matrix of **A** values, with row and column indices corresponding to evenly spaced T_1 and T_2 values. The weighing of the regularization function is determined by a parameter α, chosen by the user. In this model, any element $R_{i,j}$, i =1..*NSA*, j =1..*NSE* in the response surface **R** is assumed to follow the function:

$$R_{i,j} = f(\mathbf{A},\mathbf{T}_1,\mathbf{T}_2)_{i,j} + E_{i,j} \quad \text{with}$$
$$f(\mathbf{A},\mathbf{T}_1,\mathbf{T}_2)_{i,j} = \sum_m^M \sum_n^N A_{m,n} e^{(-(1/T_{1,n})g_i - (1/T_{2,m})t_j)} \tag{9}$$

The regularized least squares function to be minimized now takes on the form:

$$SS_{res} = \sum_{i}^{NSA}\sum_{j}^{NSE}(f_{i,j} - R_{i,j})^2 + \frac{1}{\alpha^2}\left\{\sum_{m=2}^{M-1}\sum_{n}^{N}(2A_{m,n} - A_{m+1,n} - A_{m-1,n})^2 + \sum_{m}^{M}\sum_{n=2}^{N-1}(2A_{m,n} - A_{m,n+1} - A_{m,n-1})^2\right\} \qquad (10)$$

The first term in this equation is a standard least squares term, while the second term headed by $1/\alpha^2$ is a regularization term which adjusts the curvature of the solution in T_1–T_2 space. The first term inside the regularization term covers the second derivatives along the rows of **A**, while the second term inside the regularization term covers the second derivatives along the columns of **A**. Adjusting α will thus move the solution towards the optimum leasts squares solution (large α) or towards smoothing the T_1–T_2 distribution. The actual implementation by Godefroy *et al*. will be denoted the NNLS (nonnegative least squares) algorithm in the following algorithm comparison.

Relevant criteria when comparing inverse Laplace transform algorithms are a good model fit, parsimony of the model and model robustness to noise in the data. In the present study, these criteria where studied as follows:

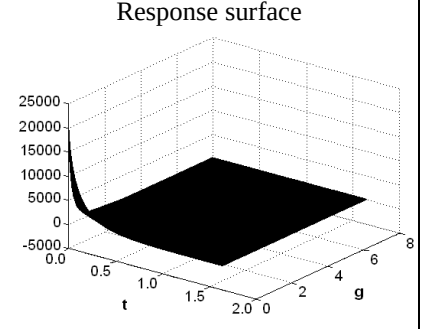
- Parsimony: Compare *NCO* in the ANAHESS model with minimal *BIC* with the number of elements in the **A** matrix in equation 15 for which $A_{m,n}$ is not equal to zero.
- Good model fit: Compare SS_{res} in equation 3 with SS_{res} in equation 10.
- Robustness to noise: Add noise to data sets and see how much and in what way the ANAHESS and NNLS results change.

The 2D-ILT results from ANAHESS and NNLS are sparse nonnegative matrices or histograms. Such sparse structures are called signatures in image analysis. A commonly used metric when comparing signatures is the Earth mover's distance (*EMD*), first introduced by French mathematician Gaspard Monge in 1781. The Earth mover's distance between two matrices is based on the following intuitive idea: Consider each nonzero element in the two matrices as heaps of earth, the size of which is determined by the size of the element. The *EMD* is then the minimal amount of work (earth mass x element distance) needed to transform the first matrix into the second. If the calculated distance between matrix elements is a true metric, then *EMD* will be a true metric. *EMD* may be calculated using linear programming, see for instance Rubner *et al*. [18]. A counterintuitive property of *EMD* is that if large elements in one of the matrices cover smaller elements in the other, then their contribution to *EMD* will be zero. Therefore, in the following algorithm comparison, all the matrices to be compared were scaled so that the sum of all elements was unity.

4.1. A Comparison of Exponential Fit Algorithms on Experimental Data

The NMR data in question were obtained from second order relaxation experiments done on a set of carbonate rock cores. The objective was to determine rock core properties such as pore sizes and porosity from NMR data. One experiment from this series is presented in table 1.

Table 1. Settings used in a second order NMR relaxation experiment on a rock core.



Core diameter	1.5 inches
Temperature during experiment	308 K
Instrument provider	Resonance[1]
Laboratory location	Anvendt teknologi[2]
Frequency	12 MHz
$g_{min}..g_{max}$	0.0010 .. 8.0 s
NSA	20
$t_{min}..t_{max}$	0.000212 .. 1.6415 s
NSE	8192

1. Resonance Instruments, now Oxford Instruments , Tubney Woods, Abingdon, Oxfordshire, OX13 5QX, UK.
2. Anvendt Teknologi AS, Munkvollveien 56, N-7022 Trondheim, Norway.

All in all, the set of plugs consisted of three dolomite samples and three calcite samples. The NMR measurements from each of the six samples were subjected to 2D-ILT analysis using both ANAHESS and NNLS. In the following, these calculations are labelled parallel 0. Then artificial noise was added to each of the raw data matrices, creating 10 parallels labelled 1,2,..10 for each rock core sample. The noise had a normal distribution with a variance equal to 1% of the maximum measurement in the response surface. These parallels were then subjected to ANAHESS and NNLS calculations, and the results studied.

In the NNLS calculations, an **A** resolution of 40 x 20 was chosen. Also, a succession of α values of 10^{11}, 10^{12}, 10^{13} and 10^{14} or 10^{10}, 10^{11}, 10^{12}, 10^{13} and 10^{14} was applied, since these choices captured the flattening out of SS_{res} as a function of α. In the ANAHESS calculations, the model with the lowest *BIC* was used consistently. Figure 1 shows *BIC* as a function of *NCO* for the first dolomite sample, parallel 0 (original data) and parallel 8 (the eighth parallel having added noise). The optimal model according to the *BIC* function is the one with *NCO* = 8 and *NCO* = 6, respectively. Figure 2 shows the two accompanying ANAHESS 2D-ILT plots. The results from all six samples are given in table 2. The corresponding NNLS calculations on parallel 0 and 8 are shown in figure 3. The NNLS calculations are summarized in table 3. Data for all the parallels, with NNLS and ANAHESS compared, are given in figures 4, 5 and 6.

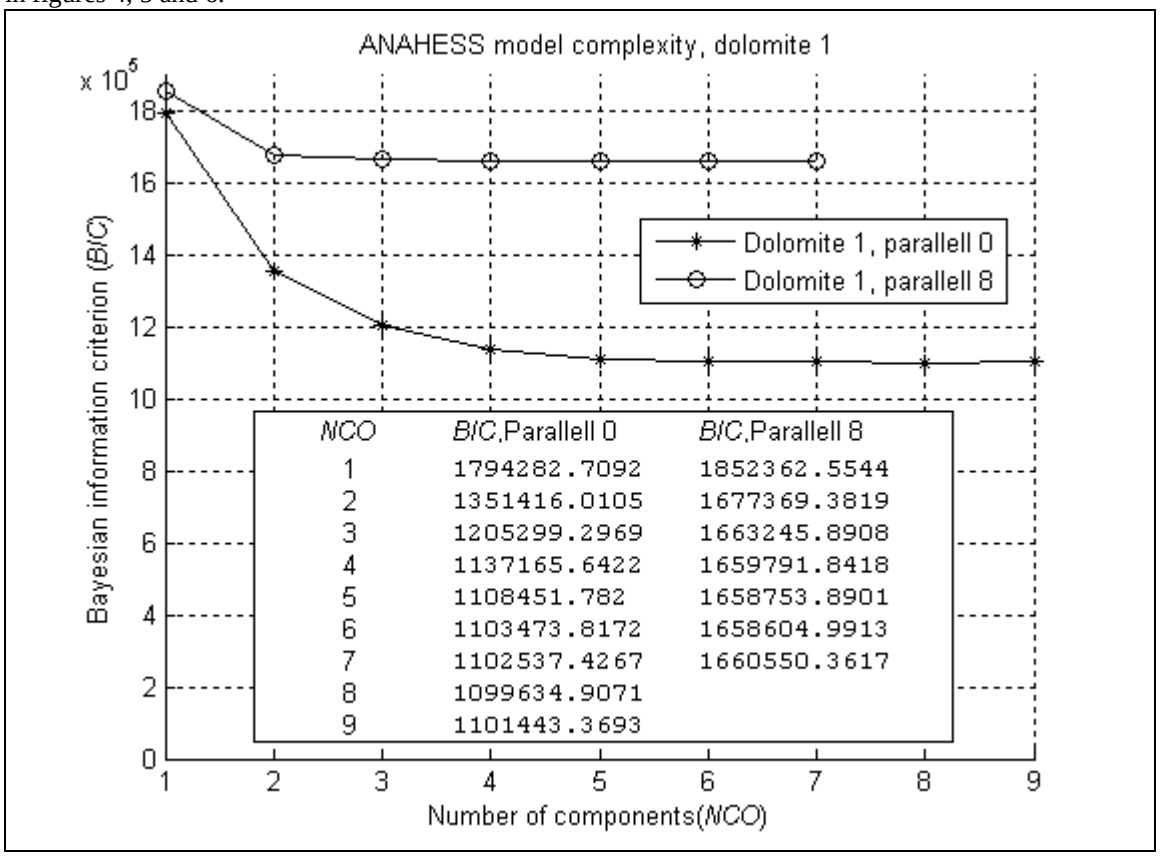


Figure 1. BIC as a function of NCO in ANAHESS. The sampleis the first dolomite, parallel 0 and parallel 8.

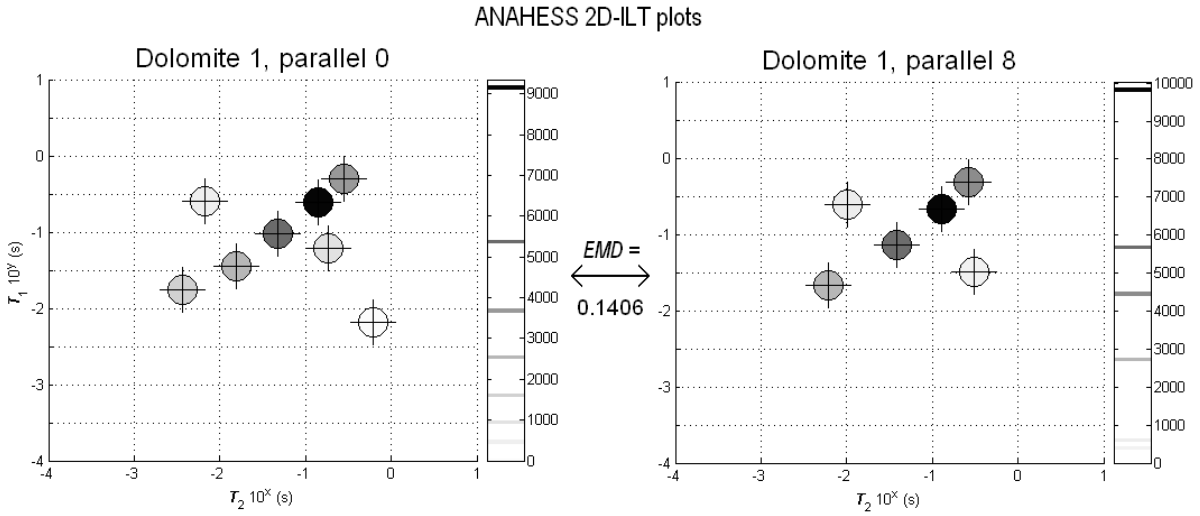


Figure 2. ANAHESS plots from the first dolomite sample, parallel 0 (NCO = 8) and parallel 8 (NCO = 6). The EMD between the two plots is 0.1406 as shown. Crosshair centers denote T_1–T_2 placement while circle greyscale denotes a_p value, as shown in sidebar.

Table 2. ANAHESS calculations on six rock core samples, with 1 original data set (parallel 0) and 10 parallels having added noise for each rock core sample. The EMD is between parallel 0 and the 10 parallels having added noise.

Sample	*NCO*, parallel 0	$SS_{res}/10^9$, parallel 0
Dolomite 1	8	0.1344
Dolomite 2	9	0.1005
Dolomite 3	6	0.1186
Calcite 1	6	0.2447
Calcite 2	6	0.1191
Calcite 3	7	0.1290

Sample	*NCO*, excl. parallel 0, 10 parallels Lowest/mean/highest			$SS_{res}/10^9$, excl. parallel 0, 10 parallels Lowest/mean/highest			*EMD* from parallel 0, 10 parallels Lowest/mean/highest		
Dolomite 1	6	6.4	8	4.0406	4.0557	4.0768	0.0544	0.1455	0.1779
Dolomite 2	4	5.8	6	3.6270	3.6377	3.6625	0.0539	0.0802	0.1532
Dolomite 3	5	5.3	6	3.9528	3.9761	4.0047	0.0282	0.0473	0.0729
Calcite 1	5	5.6	6	3.5286	3.5446	3.5936	0.0122	0.0505	0.2942
Calcite 2	4	5.1	6	2.9609	2.9744	2.9900	0.0466	0.0713	0.1059
Calcite 3	4	6.7	8	3.2239	3.2404	3.2681	0.0174	0.0750	0.1385

Table 3. NNLS calculations on six rock core samples, with 1 original data set (parallel 0) and 10 parallels having added noise for each rock core sample .The EMD is between parallel 0 and the 10 parallels having added noise.

Sample	α	*NC*O, parallel 0	$SS_{res}/10^9$, parallel 0
Dolomite 1	10^{11}	73	1.2200
	10^{12}	61	0.1864
	10^{13}	45	0.1358
	10^{14}	36	0.1343
Dolomite 2	10^{11}	57	0.5853
	10^{12}	36	0.8705
	10^{13}	28	0.1186
	10^{14}	26	0.1042
Dolomite 3	10^{11}	51	0.4992
	10^{12}	33	0.1358
	10^{13}	22	0.1283
	10^{14}	20	0.1283
Calcite 1	10^{10}	63	1.1509
	10^{11}	47	0.2885
	10^{12}	35	0.2746
	10^{13}	24	0.2726
	10^{14}	22	0.2726
Calcite 2	10^{10}	67	1.3324
	10^{11}	50	0.2767
	10^{12}	36	0.1596
	10^{13}	30	0.1255
	10^{14}	27	0.1255
Calcite 3	10^{11}	37	1.6079
	10^{12}	29	0.2827
	10^{13}	25	0.1346
	10^{14}	24	0.1346

		*NC*O, excl. parallel 0, 10 parallels Lowest/mean/highest			$SS_{res}/10^9$, excl. parallel 0, 10 parallels Lowest/mean/highest			*EMD* from parallel 0, 10 parallels Lowest/mean/highest		
Dolomite 1	10^{11}	67	73.9	90	4.7218	5.2781	5.8704	0.0420	0.0881	0.1791
	10^{12}	48	56.3	68	4.1074	4.2683	4.6708	0.0753	0.1704	0.3087
	10^{13}	41	45.1	53	4.0377	4.0696	4.1353	0.0646	0.2548	0.4997
	10^{14}	31	38	44	4.0377	4.0533	4.0717	0.1292	0.6085	1.3778
Dolomite 2	10^{11}	44	49.3	63	4.0732	4.4197	4.8994	0.0228	0.0540	0.0854
	10^{12}	30	34.4	42	3.8410	4.1235	4.6476	0.0265	0.0483	0.0712
	10^{13}	25	29.7	36	3.6220	3.6394	3.6653	0.0260	0.0650	0.1610
	10^{14}	26	28.2	31	3.6220	3.6392	3.6649	0.0238	0.1450	0.5094
Dolomite 3	10^{11}	36	40.9	47	4.2904	4.4251	4.7601	0.0183	0.0305	0.0557
	10^{12}	24	28.7	35	3.9648	3.9982	4.0260	0.0218	0.0424	0.0798
	10^{13}	20	22.8	27	3.9628	3.9866	4.0149	0.0219	0.0476	0.0816
	10^{14}	18	21.8	25	3.9628	3.9857	4.0147	0.0216	0.0479	0.0813
Calcite 1	10^{10}	56	72.6	85	4.3539	4.4780	4.6374	0.0077	0.0290	0.0516
	10^{11}	36	51.4	65	3.5685	3.5846	3.6105	0.0140	0.0410	0.0848
	10^{12}	20	38.9	51	3.5570	3.5689	3.5874	0.0191	0.0573	0.1082
	10^{13}	20	29.7	36	3.5570	3.5685	3.5869	0.0344	0.1125	0.3290
	10^{14}	17	25.4	30	3.5569	3.5686	3.5869	0.0434	0.3757	1.0262
Calcite 2	10^{10}	48	74.7	89	3.9542	4.1785	4.5342	0.0195	0.0422	0.0883
	10^{11}	34	52.8	64	3.0497	3.1231	3.1725	0.0312	0.0579	0.1092
	10^{12}	24	38	54	2.9747	2.9950	3.0233	0.0414	0.0957	0.2478
	10^{13}	20	31.2	47	2.9677	2.9804	2.9969	0.0572	0.1366	0.3474
	10^{14}	19	27.6	36	2.9676	2.9804	2.9969	0.0680	0.3452	0.7977
Calcite 3	10^{11}	32	42.5	61	4.3295	4.8571	5.6074	0.0190	0.0302	0.0583
	10^{12}	25	32.5	46	3.2501	3.3210	3.4492	0.0161	0.0467	0.0830
	10^{13}	23	27.7	33	3.2298	3.2455	3.2657	0.0267	0.0627	0.1580
	10^{14}	23	26.3	32	3.2298	3.2453	3.2654	0.0417	0.1695	0.3860

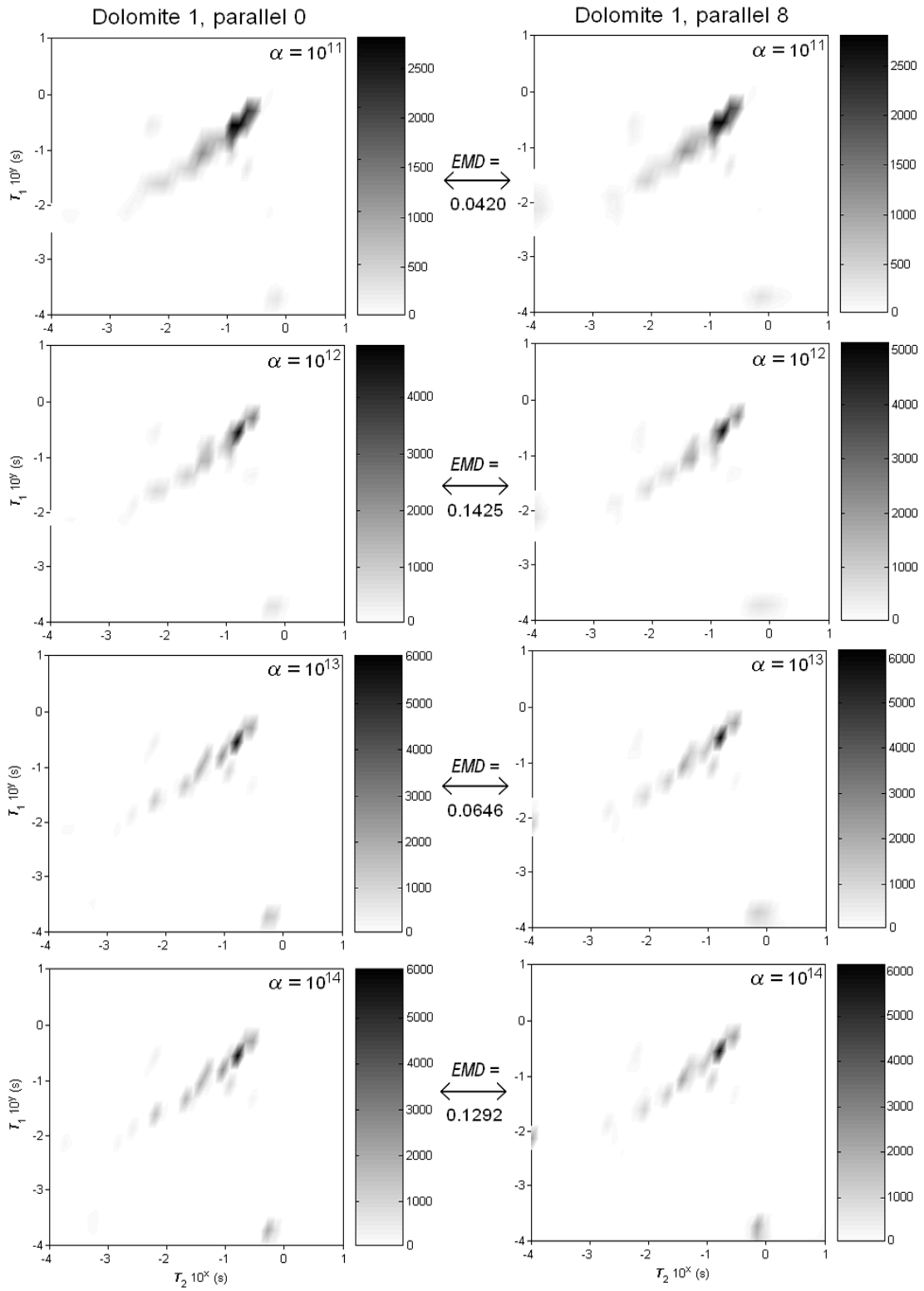


Figure 3. 2D-ILT plots from the NNLS algorithm, with 4 settings of α, run on parallel 0 and parallel 8 from the first dolomite sample. Note the EMD comparisons plot by plot.

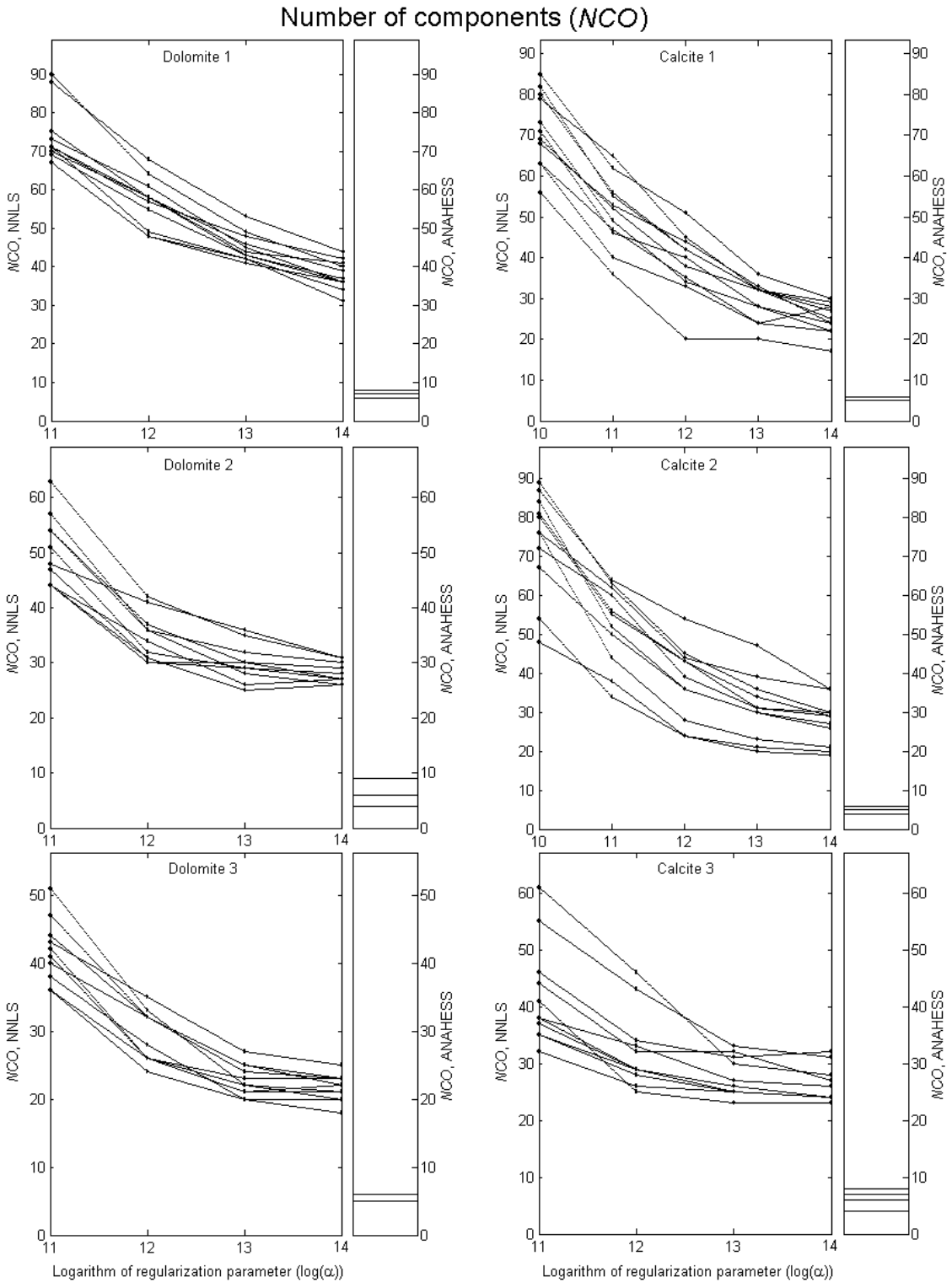


Figure 4. The number of components NCO as a function of α in NNLS, compared to the results from ANAHESS on all six rock core samples, 11 parallels per sample.

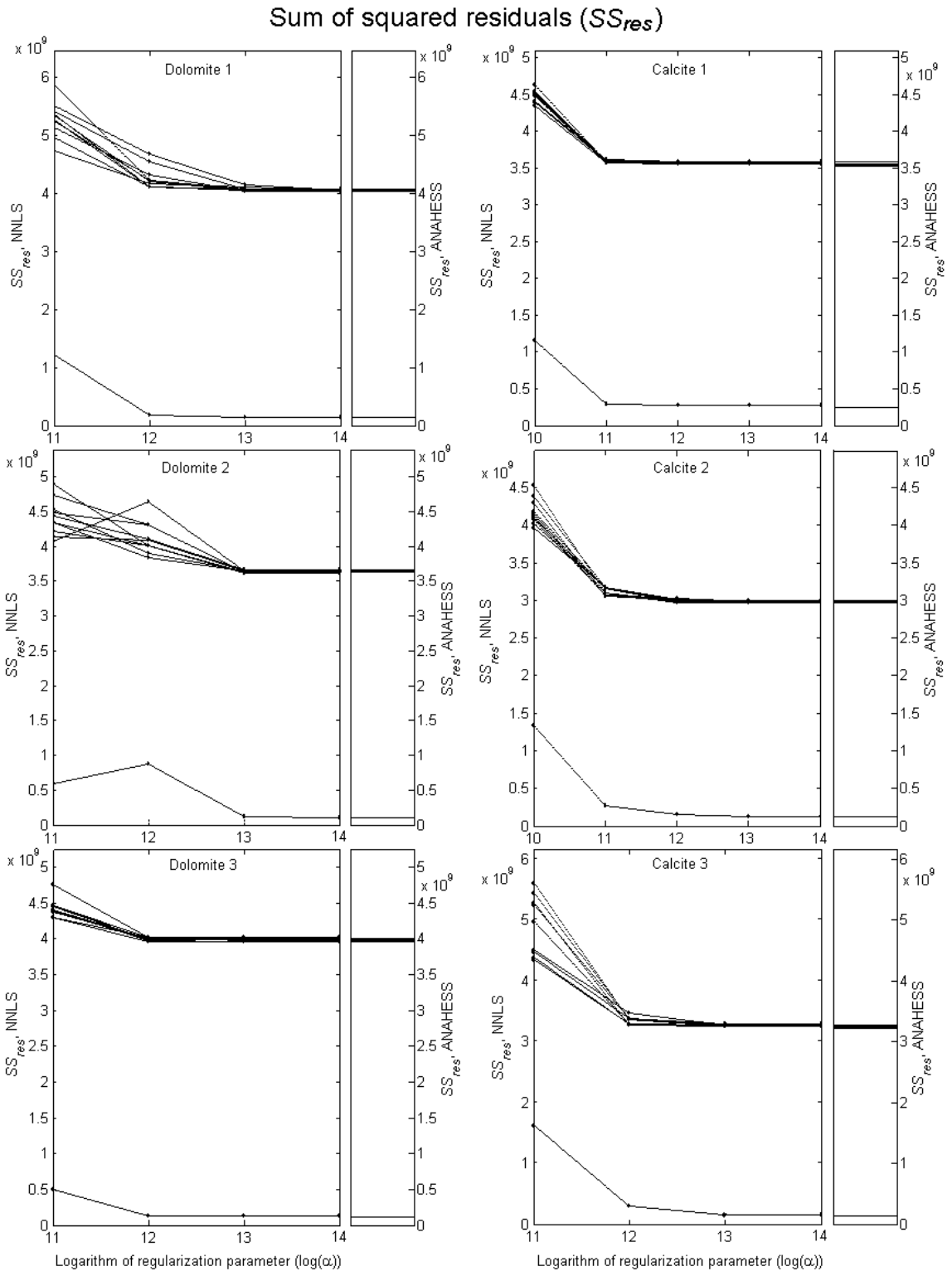


Figure 5. The sum of squared residuals (SS_{res}) as a function of α in NNLS, compared to the results from ANAHESS on all six rock core samples, 11 parallels per sample. Parallel 0 is seen below the 10 parallels that have artificial noise added.

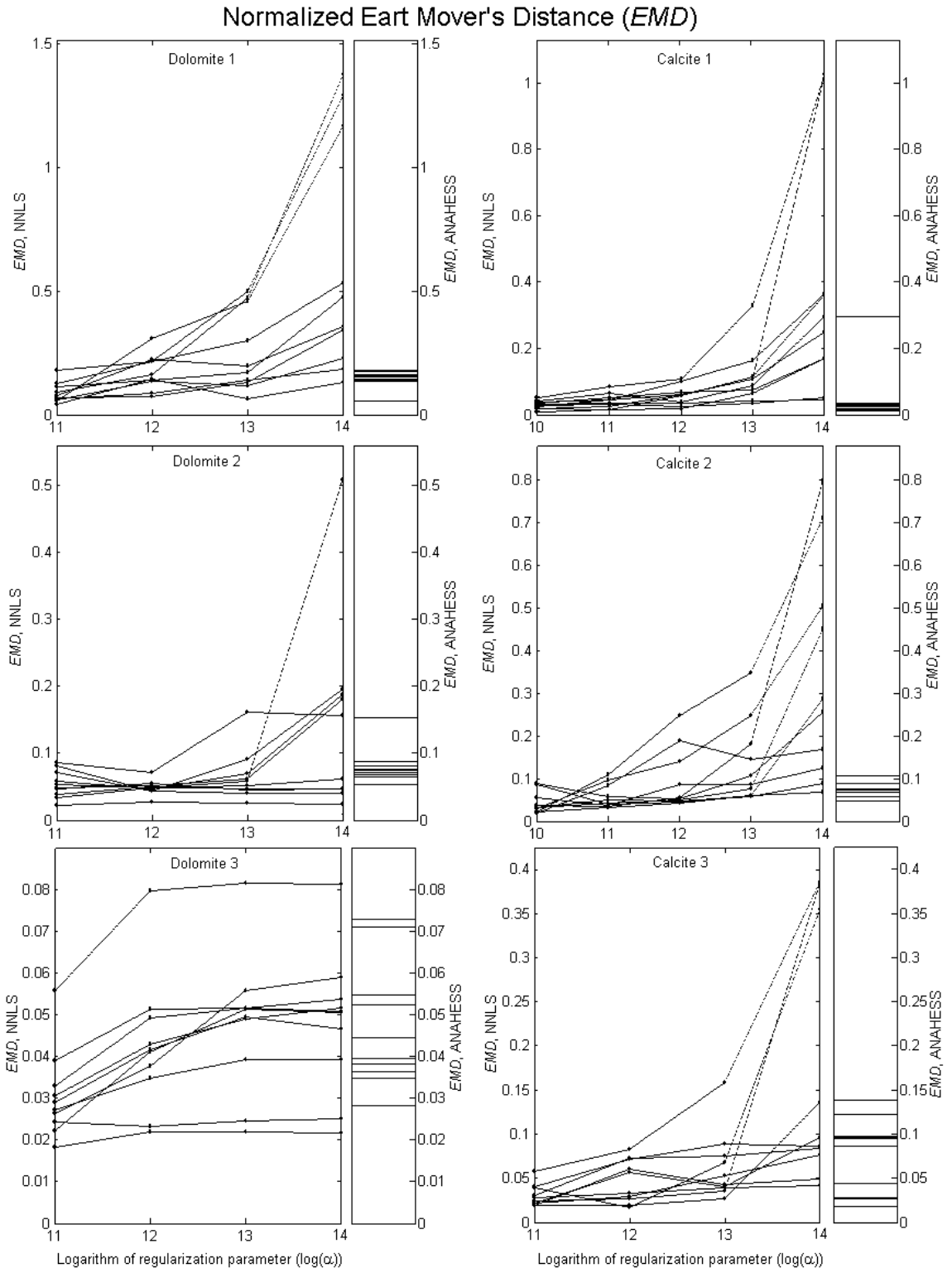


Figure 6. The normalized earth mover's distance (EMD) as a function of α in NNLS, compared to the results from ANAHESS on all six rock core samples, between 10 parallels with added noise and one unaltered parallel (parallel 0).

4.2. Conclusions From the Study on Experimental Data Sets

The most striking difference between the NNLS results and the ANAHESS results is that the number of components *NCO* is so much lower in the case of ANAHESS. In the experiments presented, *NCO* is somewhere around 4 to 12 times larger for NNLS that for ANAHESS, depending on the sample, the parallel and the choice of α. The 2D-ILT solutions from ANAHESS are parsimonious in the sense that a full model with a good fit is built using a set of few distinct solutions. The regularization term in NNLS smoothes the solution, lifting elements in the **A** matrix up from zero in the process. The result is a 2D-ILT solution with a large *NCO*.

For NNLS, the sum of squared residuals SS_{re} depends to a high degree upon the regularization parameter α. This is most striking in the case of parallel 0, in which the lowest choice of α (10^{10} for calcites 1 and 2, 10^{11} for the other four samples) gives a SS_{res} value which is 4 to 10 times the SS_{res} value obtained from the choice $\alpha = 10^{14}$ applied to the same rock core sample. The SS_{res} trend seen in the parallels with added noise is far less striking, but the dependence upon α is significant here as well. For ANAHESS, SS_{re} is about as good as it is for NNLS with α set to 10^{14}, and this is achieved with a much lower *NCO*.

The robustness to noise of the two algorithms was studied using the normalized earth mover's distance (*EMD*) between 2D-ILT solutions from raw data sets and 2D-ILT solutions from 10 parallels with added artificial noise. *EMD* is on average lower for NNLS than for ANAHESS when α is set to 10^{10}, but *EMD* rises sharply for NNLS when α is increased. The overall trend seems to be that ANAHESS holds its own against NNLS when it comes to *EMD* as well. The user needs to go for a significantly higher SS_{res} in order to achieve an *EMD* which is significantly better than the *EMD* found from ANAHESS. The overall conclusion is that ANAHESS is better than NNLS when it comes to parsimony and goodness of fit, and that ANAHESS competes well with NNLS in robustness to noise.

The underlying problem with NNLS seems to be twofold. The first difficulty with NNLS is the large number of components *NCO* which typically results. The second difficulty is the large number of operator inputs required, since the T_1–T_2 resolution (the dimensions of **A**), and the T_1–T_2 range are subject to operator choices as well as the regularization parameter α. All these operator inputs in combination influence the computational result in an unpredictable manner. In order to estimate a continuous T_1–T_2 distribution, NNLS will end up with a model which is more complex than there is numerical basis for. *NCO* in equation (10) will in most cases be so large that the resulting SS_{res} function will have many local minima. Thus, small changes in the settings of the parameters α, the T_1–T_2 range and **A** will lead to results that differ widely. The problem with NNLS is one of overmodelling.

In contrast, ANAHESS aims for an optimal solution given a fixed choice of *NCO*. In its current implementation, a solution based on $NCO = 1$ is found first. Then, subsequent solutions with higher *NCO* values are found and compared with each other according to the Bayesian information criterion (*BIC*). This criterion balances parsimony with goodness of fit, and settles for a relatively conservative estimate of *NCO*, preventing overmodelling. All in all, ANAHESS reaches a lower squared sum of residuals with fewer exponential terms than NNLS. This seems, in accordance with Occam's razor, to favour ANAHESS as the superior algorithm.

The regularization approach in the NNLS algorithm is designed to make solutions to ill posed numerical problems robust in the presence of noise. But the present study indicates that NNLS holds no significant advantage over ANAHESS. The normalized earth mover's distance (*EMD*) is a commonly used criterion for assessing similarity between sparse matrices (of which 2D-ILT solutions are examples). The *EMD* between 2D-ILT solutions from original data and 2D-ILT solutions from data having added artificial noise was calculated. The *EMD* among NNLS solutions were not significantly better than the *EMD* found among ANAHESS solutions. The conclusion is that noise does not change ANAHESS solutions more than NNLS solutions.

5. OVERALL CONCLUSIONS

A new algorithm called ANAHESS for the second order fit of a sum of exponentials has been devised and implemented. The algorithm is an inverse Hessian algorithm using analytical expressions for both the gradient and the Hessian matrix. The algorithm produces a set of distinct model components, not an approximation to a continuous distribution in T_1 and T_2. The number of components in the model is determined using the Bayesian information criterion. When compared to an established regularization method, the current MATLAB implementation of ANAHESS has performed well. ANAHESS is effective in finding simple models with a low sum of squared residuals.

6. ACKNOWLEDGEMENTS

The present work is part of VISTA project number 6334: *Improved Characterization of Rock Cores Obtained by a Combined Use of Multivariate Analysis Techniques and Magnetic Resonance (NMR)*, conducted by Statoil in close collaboration with Anvendt Teknologi A/S and the Norwegian Academy of Science and Letters. We acknowledge financial support from the Norwegian Research Council. We are also grateful to Paul Callaghan, University of Wellington, New Zealand, for the supply of the NNLS software made by Godefroy *et al*. [11, 16].

7. REFERENCES

[1] F.S.Acton, "Numerical Methods that Work". The Matematical Association of America, Washington (1990).

[2] M.D. Hürlimann, L. Burcaw, Y.-Q. Song, Quantitative characterization of food products by two-dimensional *D*-T_2 and T_1-T_2 distribution in a static gradient, *J. Colloid Interface Sci*. **297**, Issue 1, 303-311 (2006).

[3] L. Venkataramanan, Y.-Q. Song, M.D. Hürlimann, Solving Fredholm integrals of the first kind with tensor product structure in 2 and 2.5 dimensions, *IEEE Trans. Singal Process*. **50**, 1017-1026 (2002).

[4] Y.Q. Song, L. Venkataramanan, and M.D. Hürlimann, T_1-T_2 Correlation Spectra Obtained Using a Fast Two-Dimensional Laplace Inversjon, *J. Magn. Reson.* **154**, 261 (2002)

[5] G.H. Sørland, P.M. Larsen, F. Lundby, A.-P. Rudi, T. Guiheneuf, Determination of total fat and moisture content in meat using low field NMR, *Meat Sci*. **66**, 543-550 (2004).

[6] P.J. McDonald, J.-P. Korb, J. Mitchell, L. Monteilhet, Surface relaxation and chemical exchange in hydrating cement paste: A two-dimensional NMR relaxation study, *Phys. Rev. E* **72**, 011409 (2005).

[7] H. Toft Pedersen, R. Bro, S. Balling Engelsen, Towards Rapid and Unique Curve Resolution of Low-Field NMR Relaxation Data: Trilinear SLICING versus Two-Dimensional Curve Fitting, *J. Magn. Reson*. **157**, 141-155 (2002).

[8] W. Winding, A. Antalek, Direct exponential curve resolution algorithm (DECRA): A novel application of the generalized rank annihilation method for a single spectral mixture data set with exponentially decaying contribution profiles, *Chemom. Intell. Lab. Syst*. **157**, 141-155 (1997).

[9] C.H. Arns, A.P. Sheppard, R.M. Sok, M.A. Knackstedt, NMR Petrophysical Predictions on Digitized Core Images, *SPWLA 46. Annual Logging Symposium*, June 26-29 (2005).

[10] G.H. Sørland, H.W. Anthonsen, J.G. Seland, F. Antonsen, H.C. Widerøe, J.Krane, Exploring the Separate NMR Responses from Crude Oil and Water in Rock Cores, *Appl. Magn. Reson*. **26**, 417-425 (2004).

[11] S. Godefroy, B. Ryland, P.T. Callaghan, "2D Laplace Inversion Instruction Manual" (2003).

[12] W.H.Press, S.A.Teukolsky, W.T. Vetterling, B.P. Flannery, "Numerical Recipes in C", Cambridge University Press, Cambridge (1997).

[13] MATLAB, version 5.3, MathWorks Inc., Natic, Massachusetts, USA.

[14] B. Kaufmann, Fitting a Sum of Exponentials to Numerical Data, http://arxiv.org/pdf/physics/0305019 (2003).

[15] G. Schwarz, Estimating the dimension of a model. *Ann. Statist.* **6**, 461-464 (1978).

[16] S. Godefroy, B. Ryland: twodlaplaceinverse.m. TwoDLaplaceInverse, last modified 16. may 2003.

[17] C. L. Lawson, R. J. Hanson: *Solving Least Squares Problems*, Prentice-Hall, Englewood Cliffs, NJ, USA (1974).

[18] Y. Rubner, L. Guibas, C. Tomasi: The Earth Mover's Distance, Multi-Dimensional Scaling, and Color-Based Image Retrieval, Proceedings of the ARPA Image Understanding Workshop, New Orleans, LA, USA, 661-668 (1997).

[19] B. Borchers, LDLT, MATLAB Routines for Square Root Free Cholesky Factorizations http://infohost.nmt.edu/~borchers/ldlt.html.

APPENDIX: Analytic expressions of the gradient and the Hessian Matrix

In order to simplify and clarify the expressions, the matrices $\mathbf{E}_p$, p=1..*NCO* and $\mathbf{M}$ are introduced. The matrix $\mathbf{M}$ as well as each of the $\mathbf{E}_p$ matrices has dimensions *NSA* x *NSE*.

$$\begin{aligned}\mathbf{E}_p = \{E_{i,j}\}_p &= \{e^{(-(1/T_{1,p})g_i-(1/T_{2,p})t_j)}\}_p = \{e^{(-\beta_p^2 g_i-\gamma_p^2 t_j)}\}_p \\ \mathbf{M} = \{M_{i,j}\} &= \{a_0 - R_{i,j} + \sum_p^{NCO} \alpha_p^2 e^{(-\beta_p^2 g_i-\gamma_p^2 t_j)}\} \\ &= \mathbf{1}_{NSA\times 1}\mathbf{1}_{1\times NSE}\, a_0 - \mathbf{R} + \sum_p^{NCO} \alpha_p^2\, \mathbf{E}_p\end{aligned} \tag{7}$$

The function in eq. (3) to be minimized can then be written:

$$F = \sum_i^{NSA}\sum_j^{NSE}\{a_0 - R_{i,j} + \sum_p^{NCO}\alpha_p^2 (E_p)_{i,j}\}^2 = \sum_i^{NSA}\sum_j^{NSE} M_{i,j}^2 \tag{8}$$

Let $\otimes$ denote element by element matrix or vector multiplication. Then, for all indices p = 1..*NCO* and r = 1..*NCO*, define the following functions:

$$\begin{aligned}
A_0 &= 2\cdot\sum_i^{NSA}\sum_j^{NSE}\mathbf{M} & B_{0(p)} &= 4\cdot\alpha_p\cdot\sum_i^{NSA}\sum_j^{NSE}\mathbf{E}_p \\
A_{1(p)} &= 4\cdot tr(\mathbf{M}\mathbf{E}_p^{\mathrm{T}}) & B_{1(p,r)} &= 8\cdot\alpha_p\alpha_r\cdot tr(\mathbf{E}_p^{\mathrm{T}}\mathbf{E}_r) \\
A_{2(p)} &= -4\cdot\alpha_p\cdot\mathbf{g}^{\mathrm{T}} diag(\mathbf{M}\mathbf{E}_p^{\mathrm{T}}) & B_{2(p,r)} &= -8\cdot\alpha_p a_r\beta_r\cdot\mathbf{g}^{\mathrm{T}} diag(\mathbf{E}_p\mathbf{E}_r^{\mathrm{T}}) \\
A_{3(p)} &= -4\cdot\alpha_p\cdot\mathbf{t}^{\mathrm{T}} diag(\mathbf{M}^{\mathrm{T}}\mathbf{E}_p) & B_{3(p,r)} &= -8\cdot\alpha_p a_r\gamma_r\cdot\mathbf{t}^{\mathrm{T}} diag(\mathbf{E}_p^{\mathrm{T}}\mathbf{E}_r) \\
A_{4(p)} &= -8\cdot a_p\beta_p\beta_p\cdot(\mathbf{g}\otimes\mathbf{g})^{\mathrm{T}} diag(\mathbf{M}\mathbf{E}_p^{\mathrm{T}}) & B_{4(p,r)} &= 8\cdot a_p a_r\beta_p\beta_r\cdot(\mathbf{g}\otimes\mathbf{g})^{\mathrm{T}} diag(\mathbf{E}_p\mathbf{E}_r^{\mathrm{T}}) \\
A_{5(p)} &= -8\cdot a_p\gamma_p\gamma_p\cdot(\mathbf{t}\otimes\mathbf{t})^{\mathrm{T}} diag(\mathbf{M}^{\mathrm{T}}\mathbf{E}_p) & B_{5(p,r)} &= 8\cdot a_p a_r\gamma_p\gamma_r\cdot(\mathbf{t}\otimes\mathbf{t})^{\mathrm{T}} diag(\mathbf{E}_p^{\mathrm{T}}\mathbf{E}_r) \\
A_{6(p)} &= 8\cdot a_p\beta_p\gamma_p\cdot\mathbf{g}^{\mathrm{T}}(\mathbf{M}\otimes\mathbf{E}_p)\mathbf{t} & B_{6(p,r)} &= 8\cdot a_p a_r\beta_p\gamma_r\cdot\mathbf{g}^{\mathrm{T}}(\mathbf{E}_p\otimes\mathbf{E}_r)\mathbf{t}
\end{aligned} \tag{9}$$

$$C_{0(p)} = 2\cdot NSA\cdot NSE$$

$$C_{1(p)} = -4\cdot a_p\beta_p\cdot\sum_j^{NSE}\mathbf{g}^{\mathrm{T}}\mathbf{E}_p \qquad C_{2(p)} = -4\cdot a_p\gamma_p\cdot\sum_i^{NSA}\mathbf{E}_p\mathbf{t}$$

With the above functions defined, the gradient ∇ and the Hessian matrix $\mathbf{H}$ may be calculated as follows:

$$\nabla = \begin{array}{|l|}\hline \dfrac{\partial F}{\partial a_0} = A_0 \\ \hline \dfrac{\partial F}{\partial \alpha_p} = \alpha_p A_{1(p)} \\ \hline \dfrac{\partial F}{\partial \beta_p} = \alpha_p \beta_p A_{2(p)} \\ \hline \dfrac{\partial F}{\partial \gamma_p} = \alpha_p \gamma_p A_{3(p)} \\ \hline \end{array} \tag{10}$$

$$\mathbf{H} = \begin{array}{|l|l|l|l|}\hline \dfrac{\partial^2 F}{\partial a_0 \partial a_0} = C_0 & \dfrac{\partial^2 F}{\partial a_0 \partial \alpha_p} = B_{0(p)} & \dfrac{\partial^2 F}{\partial a_0 \partial \beta_p} = C_{1(p)} & \dfrac{\partial^2 F}{\partial a_0 \partial \gamma_p} = C_{2(p)} \\ \hline \dfrac{\partial^2 F}{\partial \alpha_p \partial a_0} = \dfrac{\partial^2 F}{\partial a_0 \partial \alpha_p} & \dfrac{\partial^2 F}{\partial \alpha_p \partial \alpha_p} = B_{1(p,p)} + A_{1(p)} \quad \dfrac{\partial^2 F}{\partial \alpha_p \partial \alpha_{r,p\neq r}} = B_{1(p,r)} & \dfrac{\partial^2 F}{\partial \alpha_p \partial \beta_p} = B_{2(p,p)} + 2\beta_p A_{2(p)} \quad \dfrac{\partial^2 F}{\partial \alpha_p \partial \beta_{r,p\neq r}} = B_{2(p,r)} & \dfrac{\partial^2 F}{\partial \alpha_p \partial \gamma_p} = B_{3(p,p)} + 2\gamma_p A_{3(p)} \quad \dfrac{\partial^2 F}{\partial \alpha_p \partial \gamma_{r,p\neq r}} = B_{3(p,r)} \\ \hline \dfrac{\partial^2 F}{\partial \beta_p \partial a_0} = \dfrac{\partial^2 F}{\partial a_0 \partial \beta_p} & \dfrac{\partial^2 F}{\partial \beta_p \partial \alpha_r} = \dfrac{\partial^2 F}{\partial a_r \partial \beta_p} & \dfrac{\partial^2 F}{\partial \beta_p \partial \beta_p} = B_{4(p,p)} + \alpha_p A_{2(p)} - A_{4(p)} \quad \dfrac{\partial^2 F}{\partial \beta_p \partial \beta_{r,p\neq r}} = B_{4(p,r)} & \dfrac{\partial^2 F}{\partial \beta_p \partial \gamma_p} = B_{6(p,p)} + A_{6(p)} \quad \dfrac{\partial^2 F}{\partial \beta_p \partial \gamma_{p,p\neq r}} = B_{6(p,r)} \\ \hline \dfrac{\partial^2 F}{\partial \gamma_p \partial a_0} = \dfrac{\partial^2 F}{\partial a_0 \partial \gamma_p} & \dfrac{\partial^2 F}{\partial \gamma_p \partial \alpha_r} = \dfrac{\partial^2 F}{\partial \alpha_r \partial \gamma_p} & \dfrac{\partial^2 F}{\partial \gamma_p \partial \beta_r} = \dfrac{\partial^2 F}{\partial \beta_r \partial \gamma_p} & \dfrac{\partial^2 F}{\partial \gamma_p \partial \gamma_p} = B_{5(p,p)} + \alpha_p A_{3(p)} - A_{5(p)} \quad \dfrac{\partial^2 F}{\partial \gamma_p \partial \gamma_{r,p\neq r}} = B_{5(p,r)} \\ \hline \end{array} \tag{11}$$

Note that ∇ has size $3NCO + 1$ while **H** has size $(3NCO + 1)$ x $(3NCO + 1)$. If **H** is not positive definite, then a diagonal matrix **D** is calculated and added to **H** in order to make **H** positive definite. The calculation of **D** is done using a square root free Cholesky factorization developed by B. Borchers [19].