

Regularized Gauss-Newton Method using Compressed Jacobian Matrix for Controlled Source Electromagnetic Data Inversion

Maokun Li, Aria Abubakar and Tarek M. Habashy, Schlumberger-Doll Research, Cambridge, MA, USA

SUMMARY

In this paper we propose an approach to improve the efficiency of the regularized Gauss-Newton inversion algorithm by using an adaptive cross approximation (ACA) technique. We apply the ACA technique to decompose the Jacobian matrix into two smaller rectangular matrices. In this way we improve the efficiency of the Gauss-Newton method on both memory requirements and CPU time. The improvement increases when we deal with large data sets with a large number of transmitters, receivers and frequencies. To demonstrate the improvements introduced by this method we present results of both synthetic and field data inversions for controlled source electromagnetic surveys.

INTRODUCTION

The marine controlled-source electromagnetic (CSEM) technology has the potential of providing useful information in applications such as off-shore oil exploration. With a horizontal electric dipole as a transmitter towed by a ship and multi-component electromagnetic receivers placed on the seafloor, this method has been applied in several field surveys. The high contrast in resistivity between saline-filled rocks and hydrocarbons, makes this method well-suited for detecting resistive hydrocarbon reservoirs (see Constable et al. (1986); MacGregor and Sinha (2000); Ellingsrud et al. (2002)). The approach initially employed is based on comparing the electric field amplitude as a function of the transmitter-receiver offset with a similar measurement for a non-hydrocarbon bearing reservoir, see Srnka (1986); Chave et al. (1991); Sinha (1999). The presence of hydrocarbon raises the amplitude of the measured electric field indicating the existence and, to some degree, determining the horizontal location of the hydrocarbon zone. However, with this approach it is difficult to know the reservoir's depth and shape.

A more rigorous approach to address this type of application is the full nonlinear inversion approach, for example see Abubakar et al. (2006), Gribenko and Zhdanov (2007), Plessix and van der Sman (2007) and Abubakar et al. (2008b). In such an approach the investigation domain is subdivided into pixels and by using an optimization process, the location, the shape and the conductivity of the reservoir are reconstructed. Most of these methods use iterative schemes, in which the conductivity distribution is updated based on a search direction computed from a gradient of a cost function. Therefore, in these derivative-based approaches the Jacobian matrix plays a key role. This matrix contains information about the derivative of the simulated data with respect to the conductivities of the unknown pixels. Its size is equal to the number of measurement data times the number of unknown pixels. In the CSEM data inversion, the size of the data set and the inversion region can be

very large. Hence, the storage of the Jacobian matrix requires a huge amount of memory. This is one of the bottlenecks of using the nonlinear inversion approaches. Moreover, because the Jacobian matrix is a dense matrix, the arithmetic operations of a matrix-vector multiplication can be very expensive as the size of the Jacobian matrix increases.

Currently, in order to be able to process the data, we usually invert only a subset of the data. However, the choice of this subset is often based on the experience of the data interpreters. Thus, different interpreters can use different subsets and hence, they obtain different inversion results. Furthermore important data points can easily be missed. An alternative way is to compress the Jacobian matrix. This is based on the fact that the electromagnetic field has a limited spatial bandwidth (see Bucci and Franceschetti (1987)). In real applications, we can use physical-based techniques such as the fast multiple method (Chao et al. (2003)) and the fast Fourier transform method (Bleszynski et al. (1996)). These methods can efficiently represent the field with minimum memory cost, but highly rely on the kernel of the integral involved in the matrix construction. Other methods compress the matrix through direct numerical calculations, such as a singular value decomposition (SVD) (Canning and Rogovin (2003)) or QR decomposition (Gope and Jandhyala (2004)). These methods usually compress a matrix through truncating its spectrum and removing smaller singular values. They are very general, but their computational cost to compute the matrix spectrum are usually high. Furthermore since in our case the Jacobian matrix is a full matrix, the use of these approaches may become prohibitive.

The adaptive cross approximation (ACA) technique introduced by Bebendorf (2000) provides an efficient and flexible way to compress a dense matrix. It has been applied to sample electromagnetic fields (Hislop et al. (2007)) and to solve integral equations (Zhao et al. (2005)). In this paper, we use the ACA technique to compress the Jacobian matrix in our regularized Gauss-Newton inversion algorithm. The details of our inversion algorithm can be found in Abubakar et al. (2008a) for the two-and-half dimensional (2.5D) geometry and in Abubakar et al. (2008b) for the three-dimensional (3D) geometry. The ACA technique is applied to convert the Jacobian matrix into two smaller rectangular matrices. This approach reduces both memory consumptions and CPU time of the Gauss-Newton inversion approach. To demonstrate the Gauss-Newton inversion using the compressed Jacobian matrix we employ inversion examples in 2.5D geometries.

This proceeding paper is organized as follows: We will first briefly describe the inversion algorithm. After that we discuss the application of the ACA algorithm on the matrix-vector multiplication in the Hessian matrix inversion process. Finally we discuss the performance of this scheme using both synthetic and field data examples.

Regularized Gauss-Newton inversion using compressed Jacobian matrix

FORMULATIONS

Inversion algorithm

The regularized Gauss-Newton inversion scheme is based on the work in Abubakar et al. (2008a). We will follow the same notation as in this paper. By using the multiplicative regularization technique, the cost function at the n -th iteration can be expressed as follows:

$$\Phi_n(\mathbf{m}) = \phi^d(\mathbf{m}) \times \phi_n^m(\mathbf{m}), \quad (1)$$

where ϕ^d is the data misfit cost function measuring the difference between the measurement data and the data generated from the reconstructed model. It is given by

$$\begin{aligned} \phi^d(\mathbf{m}) &= \frac{1}{2K} \sum_{k=1}^K \frac{\sum_{i=1}^{I_k} \sum_{j=1}^{J_{i,k}} |W_{d;i,j,k} [d_{i,j,k} - s_{i,j,k}(\mathbf{m})]|^2}{\sum_{i=1}^{I_k} \sum_{j=1}^{J_{i,k}} |W_{d;i,j,k} d_{i,j,k}|^2} \\ &= \frac{1}{2} \frac{|\mathbf{W}_d \cdot [\mathbf{d} - \mathbf{s}(\mathbf{m})]|^2}{|\mathbf{W}_d \cdot \mathbf{d}|^2}, \end{aligned} \quad (2)$$

where K is the number of frequencies, I_k is the number of transmitters at frequency k , and $J_{i,k}$ is the number of receivers for the transmitter i and at frequency k . The diagonal matrix $W_{d;i,j,k}$ is a data weighting matrix. $\phi_n^m(\mathbf{m})$ is the regularization cost function at the n -th iteration. More details on the data weighting matrix and the regularization cost function can be found in Abubakar et al. (2008a).

We use a Gauss-Newton minimization approach to solve Eq. 1 for conductivity values of every pixel represented by a vector of the model parameter \mathbf{m} . In each iteration n , a linearized equation is solved to obtain the step vector \mathbf{p}_n :

$$\mathbf{H}_n \cdot \mathbf{p}_n = -\mathbf{g}_n, \quad (3)$$

where \mathbf{H}_n is the Hessian matrix and it is approximated as follows:

$$\mathbf{H}_n \approx \mathbf{J}_n^T \cdot \mathbf{W}_d^T \cdot \mathbf{W}_d \cdot \mathbf{J}_n + \phi^d(\mathbf{m}_n) \mathbf{L}(\mathbf{m}_n). \quad (4)$$

In Eq. 4, \mathbf{J}_n is the Jacobian matrix and \mathbf{L} is the second derivative of ϕ_n^m with respect to the model parameter \mathbf{m} . The gradient vector \mathbf{g}_n is given by

$$\mathbf{g}_n = -\mathbf{J}_n^T \cdot \mathbf{W}_d^T \cdot \mathbf{W}_d \cdot [\mathbf{d} - \mathbf{s}(\mathbf{m}_n)] + \phi_n^d(\mathbf{m}_n) \mathbf{L}(\mathbf{m}_n) \cdot \mathbf{m}_n. \quad (5)$$

The unknown model parameter \mathbf{m} is updated using \mathbf{p}_n in every iteration. The iterative inversion process stops after some error criteria are satisfied. The Jacobian matrix is computed using an adjoint approach (Abubakar et al. (2008a)):

$$\frac{\partial E_{i,j,k}}{\partial \sigma_q}(\mathbf{r}_r, \mathbf{r}_s) = \int_{\Omega_q} d\mathbf{r}' \mathbf{E}_{i,k}^s(\mathbf{r}', \mathbf{r}_s) \cdot \mathbf{E}_{j,k}^r(\mathbf{r}', \mathbf{r}_r), \quad (6)$$

where $E_{i,j,k}$ is the simulated field at j -th receiver, i -th transmitter and k -th frequency, Ω_q is the area of the q -th pixel whose conductivity is σ_q ; $\mathbf{E}_{i,k}^s$ and $\mathbf{E}_{j,k}^r$ are the electric fields radiated by the i th transmitter and j th receiver respectively, both at frequency k .

In Eq. 3, the Hessian matrix is a dense square matrix. Its dimension is equal to the number of pixels N_p that can easily exceed thousands in 2D geometries and millions in 3D geometries. Direct solvers based on a LU decomposition or

a Gauss elimination are not practical to solve Eq. 3. Therefore, this linearized equation is usually solved using an iterative solver. Since the Hessian matrix is non-negative definite, we employ a conjugate gradient least-square (CGLS) iterative scheme (see Golub and Van Loan (1996)). The computational complexity of this CGLS solver is $O(MN_p^2)$ where M is the number of CGLS iterations. Since the number of measurement data N_m is usually much less than the number of unknown parameters N_p , in the CGLS scheme we do not explicitly construct the Hessian matrix. Instead we store the Jacobian matrix \mathbf{J} and construct routines to compute $\mathbf{J} \cdot \mathbf{v}$ and $\mathbf{J}^T \cdot \mathbf{v}$. The memory usage to store the Jacobian matrix is $O(N_m N_p)$ and the computational complexity of the CGLS solver using the Jacobian matrix-vector multiplication is $O(M(2N_m N_p))$. The operations in the CGLS routine using the Jacobian matrix is much smaller than the one using the Hessian matrix when $N_m \ll N_p$.

Though it is usually more efficient to use \mathbf{J} instead of \mathbf{H} , the size of \mathbf{J} can still become large as the number of measurement data points and/or the number of unknowns increases. For example in a CSEM survey, there are 100 receivers and 1000 transmitters, and each receiver records all six components of the electromagnetic fields at 5 frequencies. The total number of measurement data is 3 million. Assume the domain is gridded into 1000×100 cells. For double precision computation (8 bytes per real number), the memory usage for \mathbf{J} will be 9 TB. This is still beyond the capability of most current computers. Next, we will apply the ACA to reduce the memory usage and the cost of the matrix-vector multiplication in the CGLS iterative process.

Adaptive cross approximation

The adaptive cross approximation was first introduced by Bebendorf (2000) (also see Bebendorf and Rjasanow (2003)). It can be used to convert a dense Jacobian matrix \mathbf{J} into the product of two rectangular matrices, i.e., $\mathbf{J} = \mathbf{U}^T \cdot \mathbf{V}$. The detailed description of this algorithm can be found in Bebendorf (2000).

Let us assume $\mathbf{J} \in R^{m \times n}$, $\mathbf{U} \in R^{r \times m}$ and $\mathbf{V} \in R^{r \times n}$, where r is the number of rows of \mathbf{U} and \mathbf{V} after compression, $r \leq \min(m, n)$. We can write $\mathbf{U} = [\mathbf{u}_1^T, \mathbf{u}_2^T, \dots, \mathbf{u}_r^T]^T$, $\mathbf{u}_k \in R^{1,m}$; $\mathbf{V} = [\mathbf{v}_1^T, \mathbf{v}_2^T, \dots, \mathbf{v}_r^T]^T$, $\mathbf{v}_k \in R^{1,n}$. Then, the matrices \mathbf{U} and \mathbf{V} can be constructed as follows:

```

k = 0,    U = 0,    V = 0,    F = 0
while (k < min(m,n) and ||u_k||^2 ||v_k||^2 <= ε^2 · F)
    k = k + 1
    If (k = 1)    i_k = 1
    else    i_k = maxloc_p(|u_{k-1,p}|),    p ∈ (1,m) and p ∉ i
    end
    v_k = j_{i_k,:} - ∑_{p=1}^{k-1} u_{p,i_k} v_p
    j_k = maxloc_p(|v_{k,p}|),    p ∈ (1,n) and p ∉ j
    v_k = v_k / v_{k,j_k}
    u_k = j_{:,i_k}^T - ∑_{p=1}^{k-1} v_{p,i_k} u_p
    F = F + 2 ∑_{p=1}^{k-1} (u_p^T · u_k)(v_p^T · v_k) + ||u_k||^2 ||v_k||^2
end

```

where ε is the error threshold that determines convergence. In the above algorithm, the vector \mathbf{i} stores the indices of selected rows, and the vector \mathbf{j} stores the indices of selected columns;

Regularized Gauss-Newton inversion using compressed Jacobian matrix

“maxloc” is a function to search for the index of the largest element of a vector.

In the inversion algorithm, we apply ACA compression to the weighted Jacobian matrix at every frequency, i.e.,

$$\begin{aligned} \mathbf{W}_d \cdot \mathbf{J} &= \begin{bmatrix} \mathbf{W}_{d,1} \cdot \mathbf{J}_1 \\ \mathbf{W}_{d,2} \cdot \mathbf{J}_2 \\ \dots \\ \mathbf{W}_{d,K} \cdot \mathbf{J}_K \end{bmatrix} = \begin{bmatrix} \mathbf{U}_1^T \mathbf{V}_1 \\ \mathbf{U}_2^T \mathbf{V}_2 \\ \dots \\ \mathbf{U}_K^T \mathbf{V}_K \end{bmatrix} \\ &= \begin{bmatrix} \mathbf{U}_1^T & & & \\ & \mathbf{U}_2^T & & \\ & & \ddots & \\ & & & \mathbf{U}_K^T \end{bmatrix} \cdot \begin{bmatrix} \mathbf{V}_1 \\ \mathbf{V}_2 \\ \dots \\ \mathbf{V}_K \end{bmatrix} \\ &= \mathbf{U}^T \cdot \mathbf{V}. \end{aligned} \quad (7)$$

Therefore, we can re-write Eqs. 4 and 5 as follows:

$$\begin{aligned} \mathbf{h}_n &= \mathbf{V}^T \cdot \mathbf{U} \cdot \mathbf{U}^T \cdot \mathbf{V} + \phi^d(\mathbf{m}_n) \mathbf{L}(\mathbf{m}_n), \quad (8) \\ \mathbf{g}_n &= -\mathbf{V}^T \cdot \mathbf{U} \cdot \mathbf{W}_d \cdot [\mathbf{d} - \mathbf{s}(\mathbf{m}_n)] + \phi_n^d(\mathbf{m}_n) \mathbf{L}(\mathbf{m}_n) \cdot \mathbf{m}_n. \quad (9) \end{aligned}$$

The computational complexity of the Jacobian matrix vector multiplication in Eqs. 8 and 9 is $O(N_c(N_m + N_p))$, which can be smaller than the one in Eqs. 4 and 5 if

$$N_c < N_m N_p / (N_m + N_p). \quad (10)$$

Compared with other compression methods, ACA has several advantages. It relies on pure algebraic operations and does not depend on the kernel of the matrix. Moreover, in its construction it does not require the full uncompressed matrix. Therefore it is not necessary to store the full uncompressed matrix (the Jacobian matrix) in memory. The presumption of the ACA method is that the kernel of the matrix is asymptotically smooth. The Jacobian matrix for CSEM applications satisfies this criterion since the survey environment is very conductive and the frequencies of operation are very low. As given by Eq. 6, the Jacobian is the product of the electric fields radiated by the transmitters and receivers. Since the electric fields are smooth, then this product is also smooth.

NUMERICAL EXAMPLES

Inversion of a single reservoir

In this example, we consider a single reservoir located 2 km below the sea surface as shown in Figure 1(a). (We plot the resistivity $1/\sigma$ here in log scale, same for the following plots.) The reservoir is 8 km wide and 0.1 km thick. Its conductivity is 0.05 S/m. The conductivity of sea water is 3 S/m and the conductivity of the sea bed is 1 S/m. There are 41 transmitters uniformly located from -10 to 10 km in this survey, at a depth of 0.95 km. Twenty one receivers are positioned on the seabed separated by a distance of 1 km from one another. We inverted only inline electric fields at 0.25 Hz. Two percent of random white noise was added to the synthetic data. We use the 2.5D inversion algorithm to test the ACA performance. The inversion domain is from -10 to 10 km along horizontal direction and 1 to 3 km in the vertical direction. It is gridded into 100×35 pixels. The initial model is shown in Figure 1(b). The

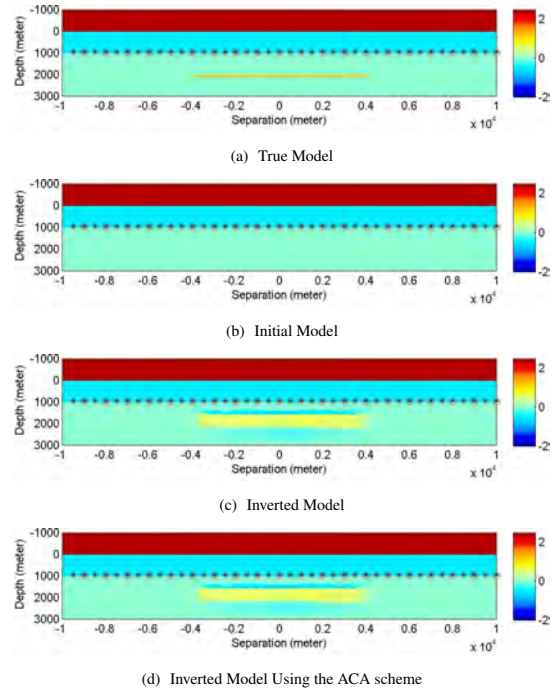


Figure 1: Inversion of single reservoir model.

weighted- L_2 norm is used in the inversion. Figure 1(c) shows the reconstructed model without using the ACA. In the inversion, the error criterion of the CGLS iterative solver is set to 10^{-3} , and the \mathbf{J} matrix has a dimension of 1600×3500 . Its memory usage is 42.7 MB. Next, we use ACA to compress the Jacobian matrix. If we set the error criterion for the ACA to be 10^{-4} , only 624 rows and columns in \mathbf{J} are used during the compression process, and the memory usage of \mathbf{U} and \mathbf{V} together is reduced to 24.3 MB. The memory reduction factor is 43.2 percent. The CPU time of each CGLS iteration is also reduced from 0.1886 seconds per iteration to 0.0484 seconds. The inverted model using the ACA scheme is shown in Figure 1(d). We note that these two inversion results are very similar.

Figure 2 shows the plots of singular values of \mathbf{J} and $\mathbf{U}^T \cdot \mathbf{V}$. The red solid-line indicates the singular values of \mathbf{J} and the blue dash-line represents the singular values of $\mathbf{U}^T \cdot \mathbf{V}$. It is observed that the compressed matrix preserves only the larger singular values in \mathbf{J} . This is consistent with the good agreement in the inversion results obtained with and without the ACA.

Figures 3 and 4 show another test result in which we compute the compressed matrix \mathbf{U} and \mathbf{V} for different numbers of transmitters. All other parameters are fixed. The error criterion for ACA is still 10^{-4} . We observe that the memory reduction ratio increases with increasing number of transmitters. This is consistent with the fact that the measured electromagnetic field has limited spatial bandwidth. Therefore the memory of \mathbf{U} and \mathbf{V} increases at a much slower rate than \mathbf{J} , and the number of rows for \mathbf{U} and \mathbf{V} does not change much. Figure 4 shows the comparison of CPU time per CGLS iteration. We also observe

Regularized Gauss-Newton inversion using compressed Jacobian matrix

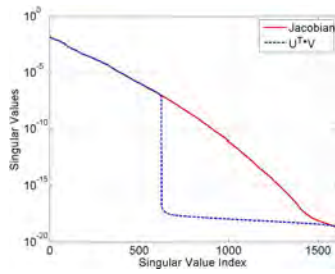


Figure 2: Comparison of singular values of the Jacobian matrix \mathbf{J} and the compressed matrix $\mathbf{U}^T \cdot \mathbf{V}$.

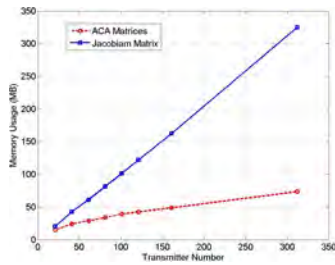


Figure 3: Comparison of memory usage of the Jacobian matrix \mathbf{J} and the compressed matrix $\mathbf{U}^T \cdot \mathbf{V}$ with increasing number of transmitters.

the improvement in efficiency by using ACA.

Inversion of Troll field data

As the next test we use the data set acquired in the Troll field in Norway (see Johansen et al. (2005)). In this survey, twenty-three receivers were located along a 13-km line on the seabed with depth about 300 m from the sea surface, and 47 transmitters were deployed at depths of about 20 m above the sea bottom. The data are inline electric fields. Two frequencies (0.25 Hz and 0.75 Hz) are used in the inversion and the total number of data points is 1516. Weighted L_2 -norm regularization is used in this inversion. The inversion result without using the ACA scheme is shown in Figure 5(b). The size of the Jacobian matrix \mathbf{J} is 3032×15872 and its memory usage is 367.15 MB. We apply the ACA scheme to \mathbf{J} with an error criterion of 10^{-3} . Only 666 columns and rows in \mathbf{J} are used. The mem-

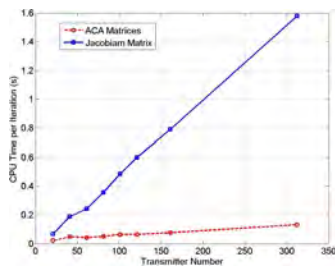


Figure 4: Comparison of CPU time per CGLS iteration using the Jacobian matrix \mathbf{J} and the compressed matrix $\mathbf{U}^T \cdot \mathbf{V}$ with increasing number of transmitters.

ory usage after ACA compression is 162.9 MB, which represents about 55 percent reduction compared to \mathbf{J} . The CPU time of one CGLS iteration reduces from 5.922 seconds to 0.8042 seconds after using ACA. The inverted model using ACA is shown in Figure 5(c). We observe that the ACA method can preserve the good inversion image while improving the efficiency and reducing the memory usage of the inversion algorithm.

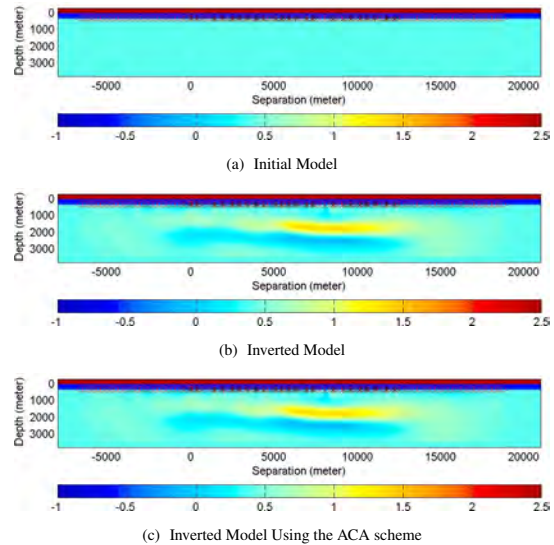


Figure 5: Inversion of the Troll field data.

CONCLUSION

In this paper we improved the efficiency of the regularized Gauss-Newton inversion by using an adaptive cross approximation (ACA) scheme. With the application of the ACA scheme to decompose the Jacobian matrix into smaller matrices, we can reduce both the memory usage and CPU time of the inversion algorithm. Moreover the memory reduction ratio increases with the increasing size of the data set. This compressed regularized Gauss-Newton inversion approach shows a great potential for handling large data sets using a manageable memory consumption.

ACKNOWLEDGMENTS

The authors would like to express their gratitude to the Troll partners, including Norsk Hydro, Statoil, Petoro, Norske Shell, Total, ConocoPhillips, Statoil, and EMGS, for access to the Troll field data. We also thank Dr. V. Druskin of Schlumberger-Doll Research, Cambridge, MA, USA and Dr. L. Knizhnerman of the Center for Geophysical Expedition, Moscow, Russia for providing the 2.5D finite-difference forward program, and Dr. Jianguo Liu of Schlumberger-Doll Research for his help on the 2.5D pixel-based inversion algorithm.

EDITED REFERENCES

Note: This reference list is a copy-edited version of the reference list submitted by the author. Reference lists for the 2009 SEG Technical Program Expanded Abstracts have been copy edited so that references provided with the online metadata for each paper will achieve a high degree of linking to cited sources that appear on the Web.

REFERENCES

- Abubakar, A., T. Habashy, V. Druskin, D. Alumbaugh, A. Zerelli, and L. Knizhnerman, 2006, Two-and-half dimensional forward and inverse modeling for marine CSEM problems: 76th Annual International Meeting, SEG, Expanded Abstracts, 750–754.
- Abubakar, A., T. M. Habashy, V. L. Druskin, L. Knizhnerman, and D. Alumbaugh, 2008a, 2.5D forward and inverse modeling for interpreting low-frequency electromagnetic measurements: *Geophysics*, **73**, no. 4, F165–F177.
- Abubakar, A., J. Liu, T. Habashy, M. Zaslavsky, and V. Druskin, 2008b, A three-dimensional multiplicative regularized non-linear inversion algorithm for cross-well electromagnetic and controlled-source electromagnetic applications: 78th Annual International Meeting, SEG, Expanded Abstracts, 584–588.
- Bebendorf, M., 2000, Approximation of boundary element matrices: *Numerische Mathematik*, **86**, 565–589.
- Bebendorf, M. and S. Rjasanow, 2003, Adaptive low-rank approximation of collocation matrices: *Computing*, **70**, 1–24.
- Bleszynski, E., M. Bleszynski, and T. Jaroszewicz, 1996, AIM: Adaptive integral method for solving large-scale electromagnetic scattering and radiation problems: *Radio Science*, **31**, 1225–1251.
- Bucci, O., and G. Franceschetti, 1987, On the spatial bandwidth of scattered fields: *Antennas and Propagation, IEEE Transactions on*, **35**, 1445–1455.
- Canning, F., and K. Rogovin, 2003, A universal matrix solver for integral-equation-based problems: *Antennas and Propagation Magazine*, **45**, 19–26.
- Chao, H.-Y., C. Lin, K. Pirapaharan, and W. C. Chew, 2003, Fast field calculation by a multilevel fast multipole algorithm for large complex radiators and scatterers: *Antennas and Propagation Society International Symposium, IEEE*, 35–38.
- Chave, A. D., S. C. Constable, and R. N. Edwards, 1991, Electrical exploration methods for the seafloor, in M. N. Nabighian, ed., *Electromagnetic methods in applied geophysics*, vol. 2: SEG, 931–966.
- Constable, S. C., C. S. Cox, and A. D. Chave, 1986, Offshore electromagnetic surveying techniques: 56th Annual International Meeting, SEG, Expanded Abstracts, 81–82.
- Ellingsrud, S., T. Eidesmo, S. Johansen, M. Sinha, L. MacGregor, and S. Constable, 2002, Remote sensing of hydrocarbon layers by seabed logging (SBL): Results from a cruise offshore angola: *Leading Edge*, **21**, 972–982.
- Golub, G. H., and C. F. Van Loan, 1996, *Matrix computations*, 3rd ed.: The Johns Hopkins University Press.
- Gope, D. and V. Jandhyala, 2004, Oct-tree-based multilevel low-rank decomposition algorithm for rapid 3-d parasitic extraction: *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, **23**, 1575–1580.
- Gribenko, A., and M. Zhdanov, 2007, Rigorous 3D inversion of marine CSEM data based on the integral equation method: *Geophysics*, **72**, no. 2, WA73–WA84.
- Hislop, G., S. Hay, and A. Hellicar, 2007, Efficient sampling of electromagnetic fields via the adaptive cross approximation: *IEEE Transactions on Antennas and Propagation*, **55**, 3721–3725.
- Johansen, S. E., H. E. F. Amundsen, T. Rsten, S. Ellingsrud, T. Eidesmo, and A. H. Bhuyian, 2005, Subsurface hydrocarbons detected by electromagnetic sounding: *First Break*, **23**, 31–36.
- MacGregor, L., and M. Sinha, 2000, Use of marine controlledsource electromagnetic sounding for sub-basalt exploration: *Geophysical Prospecting*, **48**, 1091–1106.
- Plessix, R.-E. and P. van der Sman, 2007, 3D CSEM modeling and inversion in complex geologic settings: 77th Annual International Meeting, SEG, Expanded Abstracts, 589–593.
- Sinha, M. C., 1999, Controlled source EM sounding: Survey design considerations for hydrocarbon applications: *LITHOS Science Report*, 95–101.
- Srnka, L. J., 1986, Methods and apparatus for offshore electromagnetic sounding utilizing wavelength effects to determine optimum source and detector positions: U.S. Patent.
- Zhao, K., M. N. Vouvakis, and J. Lee, 2005, The adaptive cross approximation algorithm for accelerated method of moments computations of emc problems: *IEEE Transactions on Electromagnetic Compatibility*, **47**, 763–773.