Available online at www.sciencedirect.com

SciVerse ScienceDirect

Procedia Computer Science 9 (2012) 927 – 936

Procedia Computer Science

hp-HGS strategy for inverse 3D DC resistivity logging measurement simulations

Ewa Gajda-Zagórska[a], Maciej Paszyński[a], Robert Schaefer[a], David Pardo[b], Victor Calo[c]

[a]*AGH University of Science and Technology, Krakow, Poland*
[b]*The University of the Basque Country, Bilbao, Spain and IKERBASQUE (Basque Foundation of Science)*
[c]*King Abdullah University of Science and Technology, Thuwal, Saudi Arabia*

Abstract

In this paper we present a twin adaptive strategy *hp*-HGS for solving inverse problems related to 3D DC borehole resistivity measurement simulations. The term "simulation of measurements" is widely used by the geophysical community. A quantity of interest, voltage, is measured at a receiver electrode located in the logging instrument. We use the self-adaptive goal-oriented *hp*-Finite Element Method (*hp*-FEM) computer simulations of the process of measurements in deviated wells (when the angle between the borehole and formation layers are < 90 deg). We also employ the hierarchical genetic search (HGS) algorithm to solve the inverse problem. Each individual in the population represents a single configuration of the formation layers. The evaluation of the individual is performed by solving the direct problem by means of the *hp*-FEM algorithm and by comparison with measured logging curve. We conclude the paper with some discussion on the parallelization of the algorithm.

Keywords: inverse problems, 3D DC borehole resistivity measurements, *hp* adaptive finite element method, hierarchical genetic search, parallel computing

1. Introduction

Resistivity measurements have been used during the last eighty years to quantify the spatial distribution of electrical conductivity in a reservoir. Conductivity of the rock formation is utilized to assess the material properties of the subsurface, and is routinely used by oil-companies to estimate the volume of hydrocarbons (oil and gas) existing in a reservoir. In this paper, we focus on borehole logging devices operating at low frequencies that are typically modeled numerically as zero-frequency (direct current) resistivity devices.

To improve the interpretation of results obtained with resistivity measurements, and thus, to better quantify and determine existing subsurface materials and increase hydrocarbon recovery, diverse methods have been developed to perform numerical simulations (for instance, [26, 10, 12, 7, 24, 25, 3, 19]) as well as to invert well-log measurements [2, 1, 11].

In [13, 18], we described a numerical method based on a Fourier series expansion in a non-orthogonal system of coordinates combined with a 2D self-adaptive *hp* goal-oriented Finite Element (FE) method [14, 16, 17, 15]. This Fourier-Finite-Element method was formulated and applied to direct-current (DC) and alternating-current (AC)

Email address: paszynsk@agh.edu.pl (Maciej Paszyński)
URL: http://home.agh.edu.pl/~paszynsk (Maciej Paszyński)

doi:10.1016/j.procs.2012.04.099

resistivity logging problems, and it enabled fast and accurate simulations of previously unsolved EM simulation problems in deviated wells.

In this article, we extend the above methodology to automatically solve the inverse problem using hierarchical genetic search (HGS) algorithm coupled with the self-adaptive goal oriented *hp*-FEM (*hp*-HGS). The inverse algorithm has been recently used to solve inverse problems for identification of the material data for the heat transfer problem [5] as well as linear elasticity with thermal expansion coefficient [4].

2. Forward Problem

At DC (*i.e.*, zero frequency), the electromagnetic phenomena (governed by Maxwell's equations) reduces to the so called *conductive media equation*, *i.e.*,

$$\nabla \cdot (\sigma \nabla u) = -\nabla \cdot \mathbf{J}^{imp} , \tag{1}$$

where $\sigma > 0$ is the conductivity tensor, $\mathbf{J}^{imp}$ represents the prescribed impressed electric current sources, and u is the scalar electric potential. We note that the electric field is given by $\mathbf{E} = -\nabla u$ in the case of simply connected domains.

To truncate the computational domain, in borehole logging simulations it is customary to impose zero Dirichlet boundary conditions in a large enough domain, since the solution rapidly decays as we move away from the source. In here we follow the same approach

By multiplying test function $v \in H^1_D(\Omega) = \{u \in H^1(\Omega) : u|_{\Gamma_D} = 0\}$ by equation (1), and by integrating by parts over the domain Ω, we obtain the following variational formulation after incorporating the essential (Dirichlet) boundary condition (BC):

$$\begin{cases} \text{Find } u \in u_D + H^1_D(\Omega) \text{ such that:} \\ \langle \nabla v \, , \, \sigma \nabla u \rangle_{L^2(\Omega)} = \left\langle v \, , \, \nabla \cdot \mathbf{J}^{imp} \right\rangle_{L^2(\Omega)} + \langle v \, , \, h \rangle_{L^2(\Gamma_N)} \quad \forall v \in H^1_D(\Omega) \, , \end{cases} \tag{2}$$

where $u_D \in H^1(\Omega)$ is a lift (typically $u_D = 0$) of the essential BC data u_D (denoted with the same symbol), $h = \sigma \nabla u \cdot \mathbf{n}$ is a prescribed flux on Γ_N, $\mathbf{n}$ is the unit normal outward (with respect to Ω) vector, and $u|_{\Gamma_D} = 0$ is understood in sense of traces. $\Omega \subset \mathbb{R}^3$ is assumed to be a simple connected, bounded domain with the Lipshitz boundary (see [9, 8] for necessary math details).

In our case Ω is a 3D skewed cylinder surrounding the borehole (see Figures 1, 2) crossing four geological formation layers. The direct problem can be summarized as follows: Find the electric potential u in Ω assuming the zero Dirichlet condition on the whole $\partial\Omega$ ($\Gamma_N = \emptyset$) and by given resistivities of all five layers and the borehole subdomain. The influence of probe is expressed by the assumed displacement of $\nabla \cdot \mathbf{J}^{imp}$ in the borehole subdomain. Solving the forward problem (2) for the consecutive positions of the probe we can obtain the logging curve corresponding to the assumed resistivities.

For solving the forward problem, and following the approach described in [13], we first consider the union of three (possibly rotated) cylindrical systems of coordinates defined over subdomains I, II, and III, respectively, as illustrated in Fig. 1. The change of coordinates $\boldsymbol{\zeta} = (\zeta_1, \zeta_2, \zeta_3) = \psi(\mathbf{x})$ is globally continuous and with positive Jacobian J, thus, suitable for Finite Element computations. In addition, we observe that J, as a function of ζ_2, can be represented as a linear combination of functions $1, \sin\zeta_2$, and $\cos\zeta_2$, since the change of coordinates is composed of rotations of the cylindrical system of coordinates. It is easy to see that the corresponding metric $G = J^T J$ (as a function of ζ_2) can be represented in terms of the following five Fourier basis functions: $1, \cos\zeta_2, \sin\zeta_2, \cos^2\zeta_2$, and $\sin^2\zeta_2$. Since material properties for the geometry described in Fig. 1 (deviated wells) are constant with respect to the new quasi-azimuthal direction ζ_2, and the metric can be represented exactly with only five Fourier modes, we conclude that, when using a 1D Fourier series expansion in terms of ζ_2, the corresponding stiffness matrix becomes penta-diagonal (as opposed to a dense matrix) with respect to ζ_2, leading to a dramatic reduction on the computational complexity. For details, see [13].

3. Inverse Problem

For a given reference logging curve, geometry of the formation layers and resistivities of the borehole and top and bottom formations, we seek for resistivities of the three formation layers $\omega = (\omega_0, \omega_1, \omega_2)$ (see Figure 3). The inverse problem will be formalized as the global optimization one. We are looking for each global minimizer $\hat{\omega} \in$

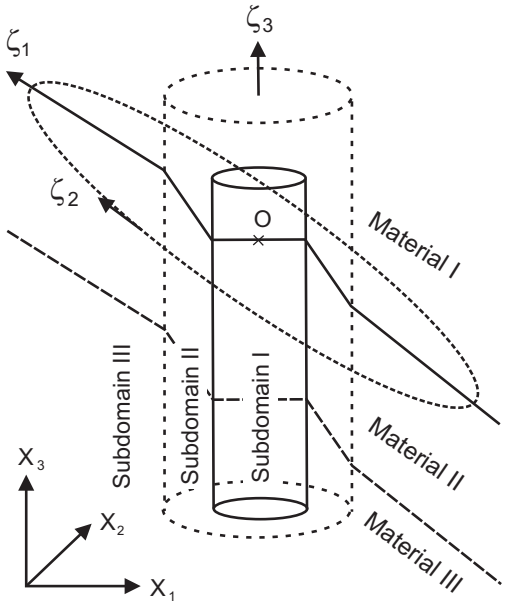

Figure 1: Cross section showing the 3D geometry of a logging instrument in a vertical well penetrating dipping layers. $\mathbf{x} = (x_1, x_2, x_3)$ represents the Cartesian system of coordinates, and $\boldsymbol{\zeta} = (\zeta_1, \zeta_2, \zeta_3)$ represents the new non-orthogonal system of coordinates. The new system of coordinates is different in each of the three sub-domains. Subdomain I corresponds to the logging instrument, subdomain II to the borehole, and subdomain III to the formation. The new system of coordinates is globally continuous, as indicated by the parameterization. Symbol 'O' indicates the origin of both systems of coordinates.

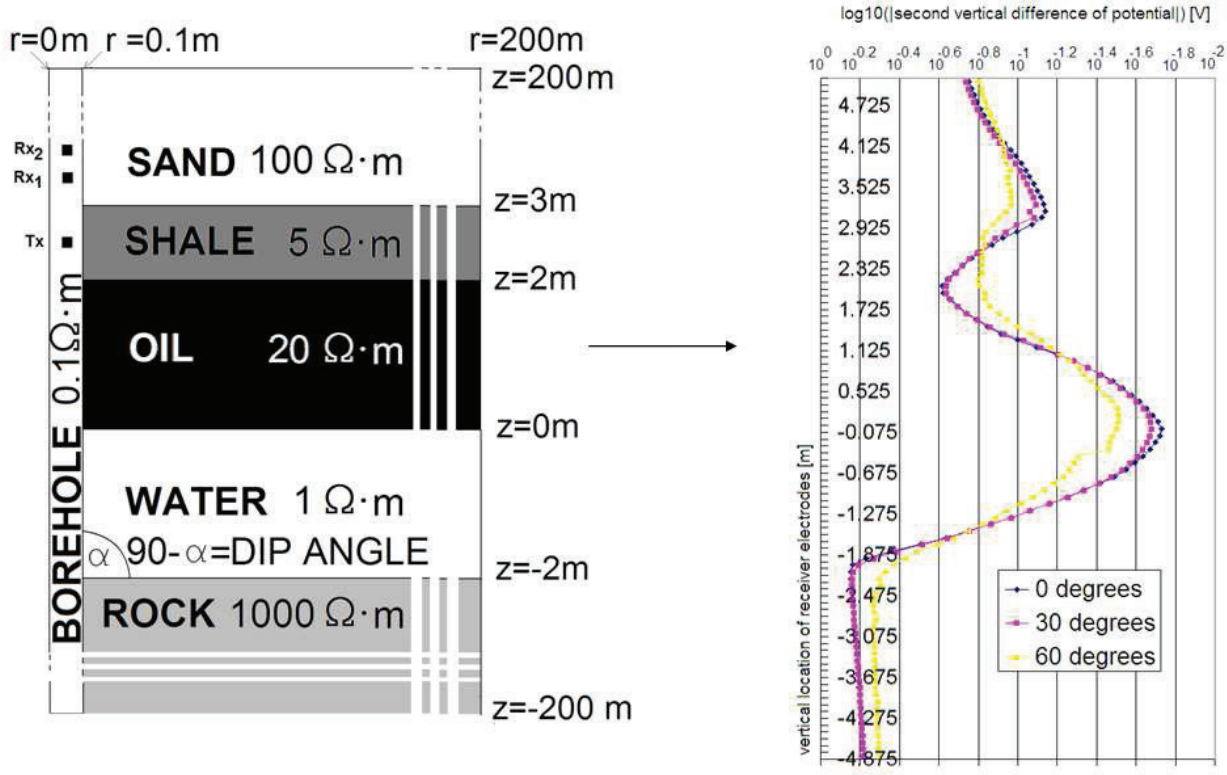

Figure 2: The direct problem consists in computations of the logging curves with assumed resistivities of the borehole and the five formation layers.

$\mathcal{D} \subset \mathbb{R}^3$ such that the corresponding, computed logging curve is closest to the reference one among all logging curves associated with $\omega \in \mathcal{D}$, where $\mathcal{D}$ stands for the set of admissible conductivities.

The reference logging curve is usually obtained from the field measurements. For testing purposes we compute this curve for the 60 degrees deviated well by using self-adaptive goal oriented *hp*-FEM algorithm with high accuracy 10^{-5}. There is a borehole with resistivity $0.1\Omega \cdot m$ and five formation layers, namely the sand with resistivity $100\Omega \cdot m$, the shale with resistivity $5\Omega \cdot m$, the oil with resistivity $20\Omega \cdot m$, the water with resistivity $1\Omega \cdot m$ and the rock with resistivity $1000\Omega \cdot m$ as in Figure 3.

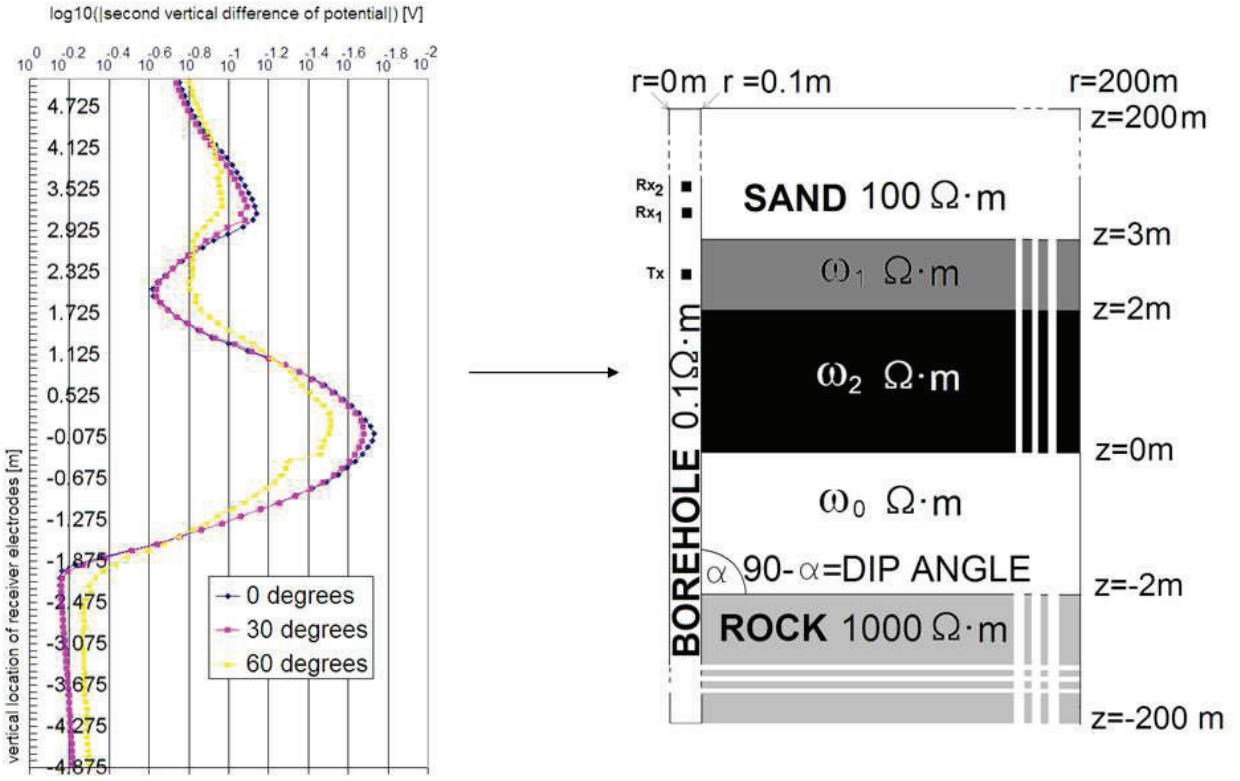

Figure 3: The inverse problem consist in finding the resistivities of the three formation layers from a given logging curve.

4. The strategy of solving inverse problem

4.1. HGS and hp–HGS

Hierarchic Genetic Strategy (HGS) introduced by Kołodziej and Schaefer [21] produces a tree-structured set of concurrent evolutionary processes (see Figure 4). The structure changes dynamically and the depth of the HGS tree is bounded by $m < +\infty$. We restrict to the case in which each evolutionary process is governed by the Simple Genetic Algorithm [23].

HGS starts with a single root deme (first order process) performing chaotic search with low accuracy. After a constant number of genetic epochs K called the *metaepoch* the root deme sprouts child-demes in the promising regions of the evolutionary landscape surrounding the best fitted individuals distinguished from the parental deme (see left panel in Figure 5). The second order demes (branches) perform more local search with higher accuracy. Third order demes (leaves) are created at the results of the second level search (see middle panel in Figure 5), and they perform local and most accurate search (see right panel in Figure 5). The evolution in existing demes continues in the second metaepoch, after which new demes are sprouted. The algorithm continues until the global stop condition is reached.

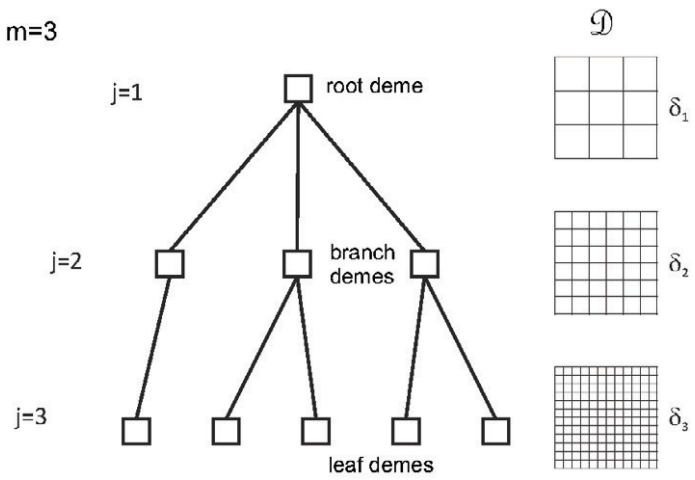

Figure 4: HGS tree and corresponding two-dimensional meshes.

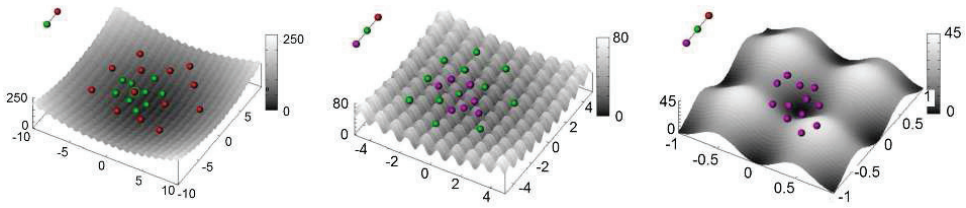

Figure 5: Left panel: root population (red) sprouting second level deme (green). Middle panel: second level population (green) sprouting leaf population (pink). Right panel: leaf population (pink) performing local search.

HGS implements two mechanisms that prevent redundancy of the search. The first one, called *conditional sprouting*, allows new demes to be sprouted only in regions which are not explored by sibling-demes (demes sprouted by the same parent). The second mechanism, called *branch reduction*, reduces demes of the same order that perform search in the common landscape region or in the regions already explored.

Different search accuracies are obtained by various encoding precisions and by manipulating the length of binary genotypes in demes at different levels. The root utilizes the shortest genotypes, while the leaves utilize the longest ones. To obtain search coherency for demes of different orders, a special kind of hierarchical nested encoding is used. Firstly, the densest mesh of phenotypes in $\mathcal{D}$ for the demes of the m-th order is defined. Afterwards, the meshes for lower order demes are recursively defined by selecting some nodes from the previous ones. The maximum diameter of the mesh δ_j associated with the demes of the order j determines the search accuracy at this level of the HGS tree (see Figure 4). The mesh parameters satisfy $\delta_m < \ldots < \delta_1$.

HGS can be coupled with the *hp*–adaptive goal–oriented FEM algorithm in order to solve the particular class of inverse, parametric problems [5]. In this strategy called *hp*-HGS the fitness of each individual is estimated with the required accuracy depending on the level of the HGS tree. Root level direct problems are solved with lowest accuracy, while leaf level direct problems are solved with highest accuracy.

4.2. Solving 3D DC resistivity logging measurement problem by hp–HGS

We performed the simulation of the 3D DC borehole resistivity measurements problem using *hp*-HGS method with $m = 3$. The parameters of the simulation are presented in Table 1. Populations sizes were selected to balance the time of evaluating a solution without losing the property of searching globally. Code length for a single parameter was 15 on the first level and was increased by 6 on the second level and by 12 in leaves. Fitness value of each candidate solution ω (resistivity vector) was evaluated as the Euclidean norm of the difference between discrete representations of the reference logging curve and the logging curve computed by the self-adaptive goal-oriented *hp*–FEM algorithm

for ω with accuracy depending on the level in HGS tree. The accuracy (see last row in Table 1) is the maximum relative error decrement in the single hp-FEM step (see e.g. [20]) applied for solving direct problem at the particular HGS level.

Table 1: Parameters of the simulation.

	level 1	level 2	level 3
Population size	12	6	4
Code length	45	63	81
Mutation rate	0.1	0.01	0.001
Crossing rate	0.5	0.5	0.5
Accuracy	0.1	0.01	0.001

After the first metaepoch performed at three levels HGS tree consists of three demes - one at each level (see Figure 6). Total execution times for the entire populations are presented in the middle panel of the Figure 6 and average execution times for a single individuals are presented in the right panel. All execution times are rounded due to clarity reasons. It is clear that the number of high-level individuals should be kept as small as possible because of their time-consuming evaluation.

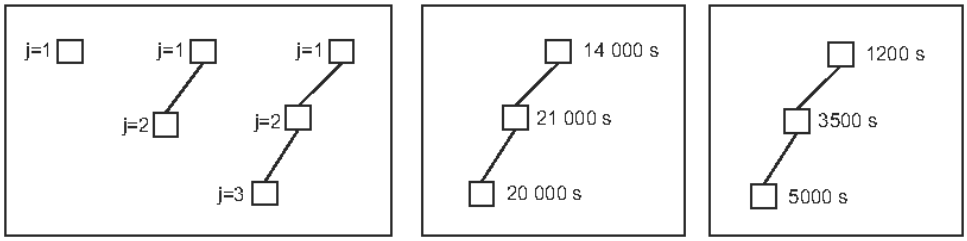

Figure 6: Metaepoch 1. Left panel: evolution of the HGS tree in the first metaepoch. Middle panel: total execution time for entire population. Right panel: average execution time for a single individual.

HGS tree after the second metaepoch is presented in Figure 7. In order to reduce the number of fitness evaluations, all calculated values are stored. If an individual remains unchanged in the next epoch or its genotype is changed into one that was already evaluated on the same level, a previously stored value is used. This approach allows a significant reduction of execution time (see i.e. first leaf population in Figure 7).

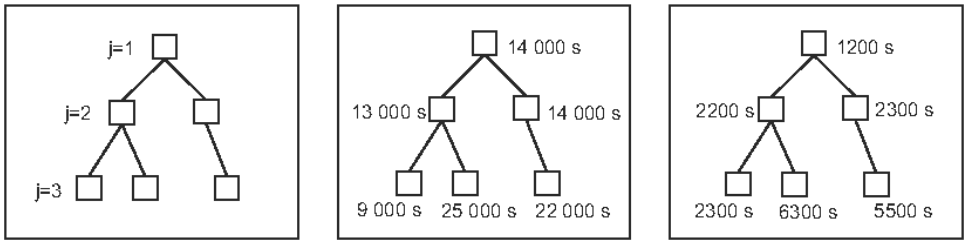

Figure 7: Metaepoch 2. Left panel: tree of populations after the second metaepoch. Middle panel: total execution time for entire population. Right panel: average execution time for a single individual.

Table 2: Computation results.

ω_1	ω_2	ω_3	error	fitness
0.995	3.605	44.166	24.20	0.10
0.760	6.289	33.791	13.85	0.14
0.750	6.345	27.674	7.79	0.16
0.789	3.578	19.166	1.66	0.20

5. Numerical results and scalability evaluation

We have implemented *hp*-HGS algorithm in Linux cluster environment. The problems was solved in 6 metaepochs. The computation was stopped when the best leaf individual fitness was less or equal 0.10. Simulation results are illustrated in Figure 8. Table 2 gathers selected four well-fitted leaf individuals. They show the potential power of *hp*-HGS in resistivity identification (the smallest error was about 1.66) and the fitness irregularity in the neighbourhood of the global minimum.

The complicated scheduling of parallel computation (strongly varying number of available processors) makes the wall time of computation hardly reliable. However it was possible to measure the serial execution time of particular tasks which can be used in further scalability analysis. First, the serial execution time can be evaluated as 929827 seconds (about 258 hours or 10 days and 18 hours).

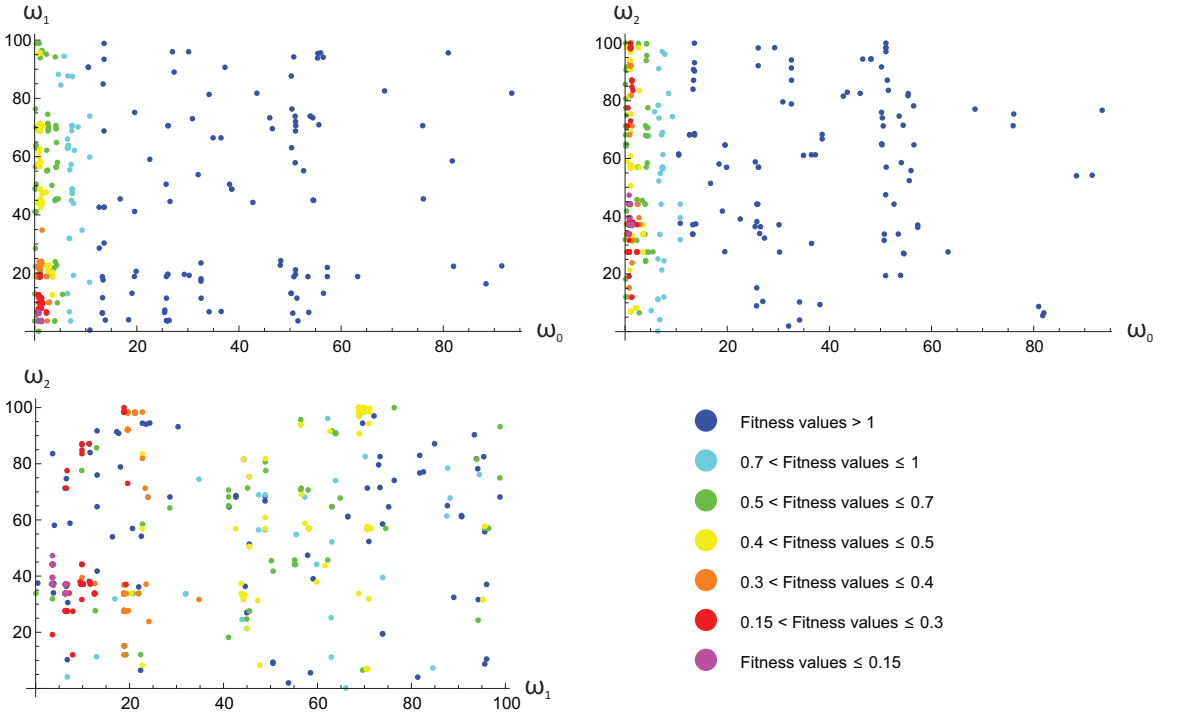

Figure 8: Scatter plots of ω_0 vs. ω_1 (top–left), ω_0 vs. ω_2 (top–right) and ω_1 vs. ω_2 (bottom–left). Different colors correspond to different fitness values.

In Figure 9 we list the total number of populations in particular epochs during the entire simulation. We can parallelize the code on the level of populations for up to 11 processors. The total execution time defined as a sum of calls of the self-adaptive goal-oriented *hp*-FEM for all individuals from all populations from particular epochs is

summarized in Figure 10. We conclude that the parallel inverse code with 11 processors will run for about 183,000 second = 51 hours (2 days and 3 hours). Obtained speedup equals 5.08 and efficiency is 45%.

To speedup the code further we can perform the computations for all individuals from all populations from a given epoch at the same time. In Figure 11 we list the total number of individuals from all populations in particular epochs. Taking into consideration the real number of fitness evaluations, we can parallelize the code on the level of individuals by using up to 30 processors. For each individual we need to call the self-adaptive goal-oriented *hp*-FEM algorithm. In Figure 12 we list execution time for the heaviest individual from all population for a given epoch. We conclude that the parallel inverse code with 30 processors will run for about 57,400 second = 16 hours. In this case we obtain speedup of 16.2 and efficiency of 54%.

To speedup the code further we can switch to the parallel version of the self-adaptive goal-oriented *hp*-FEM [6, 20]. The parallel code delivers 60% efficiency up to 128 cores (namely the computations of a single individual are 80 times faster on 128 cores). We conclude that the parallel inverse code with $30 \times 128 = 3,840$ cores will run for about $57,400/80 = 717.5$ second = 0.2 hour (12 minutes). The final speedup equals 1295.9 with efficiency of 34%.

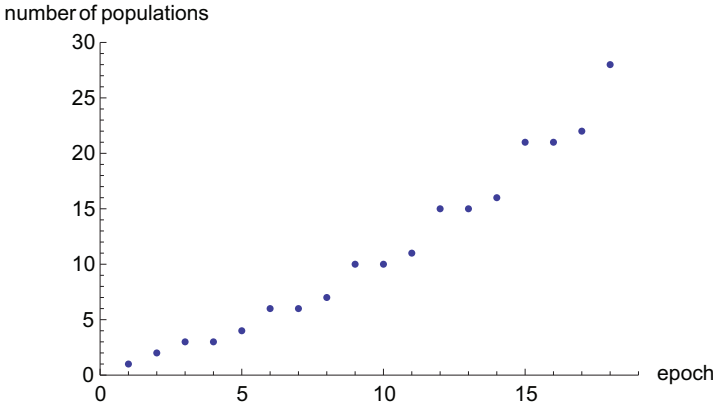

Figure 9: Number of populations in particular epochs (each metaepoch consists in several epochs).

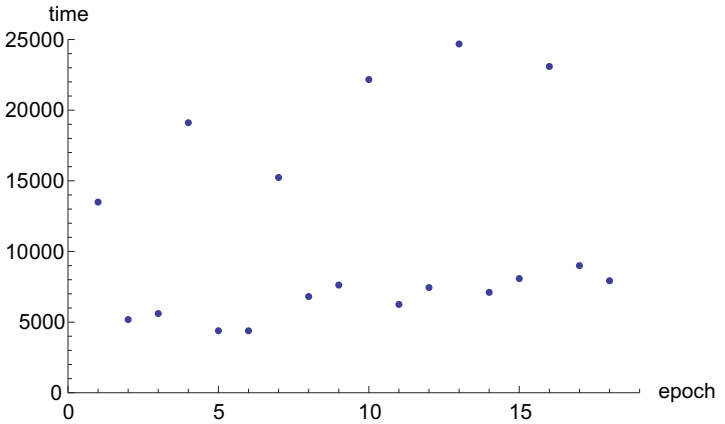

Figure 10: Total execution time for largest populations in particular epochs.

6. Conclusions

The *hp*-HGS strategy was applied and tested for the 3D DC borehole resistivity measurement simulations problem, that is of the great interest to the oil industry. The algorithm utilizes the Hierarchical Genetic Strategy interfaced with self-adaptive *hp* Finite Element Method. The inverse problem was successfully solved with the acceptable final

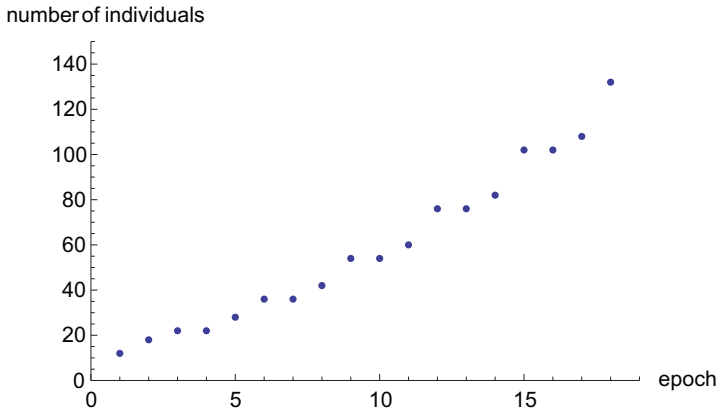

Figure 11: Number of individuals in particular epochs.

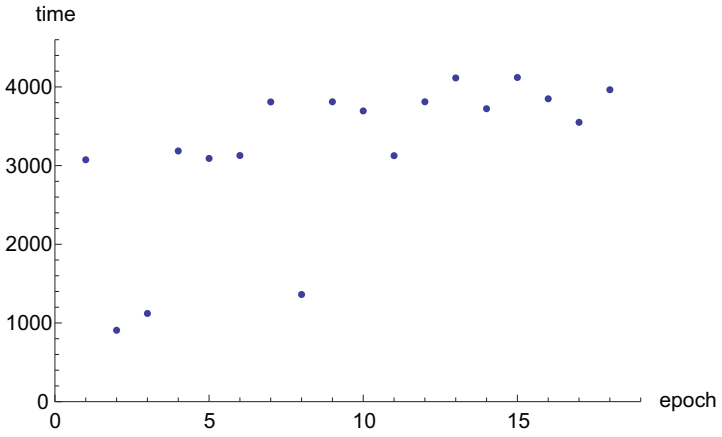

Figure 12: Total execution time for largest individual in particular epochs.

fitness 0.10 within 6 metaepochs. The simple derivation of the parallel computation time for this example show the advantage of the high level fine-grained strategy coupled with the parallel execution of the special instance of *hp*-FEM. The simple population-level coarse-grained parallel execution of *hp*-HGS would result in 51 hours while fine-grained individual-level one consumes the 16 hours. The fine-grained *hp*-HGS execution coupled with parallel *hp*-FEM reduces the computation time up to 0.2 hour. If maximum speedup is required, it can be obtained by assigning a huge number of processors that work with moderate efficiency 34 - 54% (i.e. they are partially idle). It is a result of a complicated, stochastic character of the computational tasks appearing during the *hp*-HGS run. The proper dynamic scheduling can significantly increase the efficiency by a slight loss of speedup (see e.g.[22]). Moreover, the fitness needs to be locally regularized in order to omit the non-monotonic behavior in the neighborhood of the global extremum.

Acknowledgments

The work was partially supported by the Polish MNiSW grant no. NN 519 447739. The work reported in this paper was partially funded by the Spanish Ministry of Sciences and Innovation under project MTM2010-16511.

References

[1] A. Abubakar, T.M. Habashy, V. Druskin, L. Knizhnerman, and S. Davydycheva. A 3D parametric inversion algorithm for triaxial induction data. *Geophysics*, 71:G1–G9, 2006.

[2] A. Abubakar and P. M. van den Berg. Nonlinear inversion in electrode logging in a highly deviated formation with invasion ising an oblique coordinate system. *IEEE Transactions on Geoscience and Remote Sensing*, 38:25–38, 2000.
[3] D. B. Avdeev, A. V. Kuvshinov, O. V. Pankratov, and G. A. Newman. Three-dimensional induction logging problems, part 1: An integral equation solution and model comparisons. *Geophysics*, 67:413–426, 2002.
[4] B. Barabasz, E. Gajda, S. Migórski, M. Paszyński, and R. Schaefer. Studying inverse problems in elasticity by hierarchic genetic search. In *ECCOMAS thematic conference on Inverse Problems in Mechanics of Structures and Materials*, 2011.
[5] B. Barabasz, S. Migórski, R. Schaefer, and M. Paszyński. Multi-deme, twin adaptive strategy hp-HGS. *Inverse Problems in Science and Engineering*, 19:3–16, 2011.
[6] V. Calo, D. Pardo, and M. Paszyński. Goal-Oriented Self-Adaptive hp Finite Element Simulation of 3D DC Borehole Resistivity Simulations. *Procedia Computer Science*, 4:1485–1495, 2011.
[7] S. Davydycheva, V. Druskin, and T. Habashy. An efficient finite-difference scheme for electromagnetic logging in 3D anisotropic inhomogeneous media. *Geophysics*, 68(5):1525–1536, 2003.
[8] Z. Denkowski, S. Migórski, and N.S. Papageorgiou. *An Introduction to Nonlinear Analysis: Applications*. Kluwer Academic/Plenum Publishers, 2003.
[9] Z. Denkowski, S. Migórski, and N.S. Papageorgiou. *An Introduction to Nonlinear Analysis: Theory*. Kluwer Academic/Plenum Publishers, 2003.
[10] V. L. Druskin, L. A. Knizhnerman, and P. Lee. New spectral Lanczos decomposition method for induction modeling in arbitrary 3-D geometry. *Geophysics*, 64(3):701–706, 1999.
[11] X. Lu and D. L. Alumbaugh. One-dimensional inversion of three-component induction logging in anisotropic media. *SEG Expanded Abstract*, 20:376–380, 2001.
[12] G. A. Newman and D. L. Alumbaugh. Three-dimensional induction logging problems, part 2: A finite-difference solution. *Geophysics*, 67(2):484–491, 2002.
[13] D. Pardo, V. M. Calo, C. Torres-Verdín, and M. J. Nam. Fourier series expansion in a non-orthogonal system of coordinates for simulation of 3D DC borehole resistivity measurements. *Computer Methods in Applied Mechanics and Engineering*, 197(1-3):1906–1925, 2008.
[14] D. Pardo, L. Demkowicz, C. Torres-Verdín, and M. Paszynski. Simulation of resistivity logging-while-drilling (LWD) measurements using a self-adaptive goal-oriented *hp*-finite element method. *SIAM Journal on Applied Mathematics*, 66:2085–2106, 2006.
[15] D. Pardo, L. Demkowicz, C. Torres-Verdín, and M. Paszynski. A goal oriented *hp*-adaptive finite element strategy with electromagnetic applications. Part II: electrodynamics. *Computer Methods in Applied Mechanics and Engineering.*, 196:3585–3597, 2007.
[16] D. Pardo, C. Torres-Verdín, and L. Demkowicz. Simulation of multi-frequency borehole resistivity measurements through metal casing using a goal-oriented *hp*-finite element method. *IEEE Transactions on Geosciences and Remote Sensing*, 44:2125–2135, 2006.
[17] D. Pardo, C. Torres-Verdín, and L. Demkowicz. Feasibility study for two-dimensional frequency dependent electromagnetic sensing through casing. *Geophysics*, 72:F111–F118, 2007.
[18] D. Pardo, C. Torres-Verdín, M. J. Nam, M. Paszynski, and V. M. Calo. Fourier series expansion in a non-orthogonal system of coordinates for simulation of 3D alternating current borehole resistivity measurements. *Computer Methods in Applied Mechanics and Engineering*, 197:3836–3849, 2008.
[19] D. Pardo, C. Torres-Verdín, and M. Paszynski. Simulation of 3D DC borehole resistivity measurements with a goal-oriented *hp* finite element method. Part II: Through casing resistivity instruments. *Computational Geosciences*, 12(1):83–89, 2008.
[20] M. Paszyński, D. Pardo, C. Torres-Verdn, L. Demkowicz, and Calo V. A parallel direct solver for the self-adaptive hp Finite Element Method . *Journal of Parallel and Distributed Computing*, 70:270–281, 2010.
[21] R. Schaefer R. and J. Kołodziej. Genetic search reinforced by the population hierarchy. In *Foundations of Genetic Algorithms 7*, pages 383–399. Morgan Kaufman Publisher, 2003.
[22] P. Uhruski, M. Grochowski, and R. Schaefer. A Two-Layer Agent-Based System for Large-Scale Distributed Computation. *Computational Intelligence*, 24:191–212, 2008.
[23] M. D. Vose. *The Simple Genetic Algorithm*. MIT Press, 1999.
[24] T. Wang and S. Fang. 3-D electromagnetic anisotropy modeling using finite differences. *Geophysics*, 66(5):1386–1398, 2001.
[25] T. Wang and J. Signorelli. Finite-difference modeling of electromagnetic tool response for logging while drilling. *Geophysics*, 69(1):152–160, 2004.
[26] J. Zhang, R. L. Mackie, and T. R. Madden. 3-D resistivity forward modeling and inversion using conjugate gradients. *Geophysics*, 60:1312–1325, 1995.