

Research Article

Three-Dimensional Induced Polarization Parallel Inversion Using Nonlinear Conjugate Gradients Method

Huan Ma,^{1,2} Handong Tan,^{1,2} and Yue Guo³

¹Key Laboratory of Geo-Detection, China University of Geosciences, Ministry of Education, Beijing 100083, China

²School of Geophysics and Information Technology, China University of Geosciences, Beijing 100083, China

³Exploratory Drilling Corporation Well Logging Company, Daqing, Heilongjiang 163000, China

Correspondence should be addressed to Handong Tan; thd@cugb.edu.cn

Received 19 March 2015; Accepted 27 April 2015

Academic Editor: Sergio Preidikman

Copyright © 2015 Huan Ma et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Four kinds of array of induced polarization (IP) methods (surface, borehole-surface, surface-borehole, and borehole-borehole) are widely used in resource exploration. However, due to the presence of large amounts of the sources, it will take much time to complete the inversion. In the paper, a new parallel algorithm is described which uses message passing interface (MPI) and graphics processing unit (GPU) to accelerate 3D inversion of these four methods. The forward finite differential equation is solved by ILU0 preconditioner and the conjugate gradient (CG) solver. The inverse problem is solved by nonlinear conjugate gradients (NLCG) iteration which is used to calculate one forward and two “pseudo-forward” modelings and update the direction, space, and model in turn. Because each source is independent in forward and “pseudo-forward” modelings, multiprocess modes are opened by calling MPI library. The iterative matrix solver within CULA is called in each process. Some tables and synthetic data examples illustrate that this parallel inversion algorithm is effective. Furthermore, we demonstrate that the joint inversion of surface and borehole data produces resistivity and chargeability results are superior to those obtained from inversions of individual surface data.

1. Introduction

IP methods are important in geophysical electrical surveys. Surface exploration is used for detecting metallic and non-metallic minerals, using various observational techniques. Borehole-surface, surface-borehole, and borehole-borehole exploration methods are also widely used in mining field contexts and give improved results for the prospecting for ores, oil, and gas.

Many previous resistivity and IP inversion methods have been developed, including 2D least-squares inversion [1], a 3D resistivity inversion using alpha center method [2], many linear and nonlinear IP inversion methods [3, 4], and a 3D IP conjugate inversion method [5].

Since 3D problems are very much larger than 1D and 2D problems, their solution is correspondingly more costly in computer time. Parallel MPI and GPU algorithms are generally used for numerical calculation. MPI is a library that is called by Fortran77, Fortran90, C, and C++ software and serves to communicate between processes. It has two basic

patterns: the equal pattern and the principal-subordinate pattern [6].

GPUs are parallel processors with a large number of computation units and are superior to CPUs in both processing capability and memory bandwidth. They are cheaper and consume less power, and they are therefore used for large-scale, high-performance computation work [7, 8]. Programs are written in the familiar C, C++, Fortran, or an ever-expanding list of supported languages [9]. CULA is a set of GPU-accelerated linear algebra libraries utilizing the NVIDIA CUDA parallel computing platform to dramatically improve the speed of advanced mathematical computation. In other words, to use these functions an NVIDIA GPU with CUDA support is required. The library contains CULA Dense and CULA Sparse.

Solution is very slow for forward and “pseudo-forward” equations using the nonlinear conjugate gradients method. Therefore, a set of 3D IP forward and inversion parallel algorithms are needed for surface, borehole-surface, surface-borehole, and borehole-borehole techniques. This paper

introduces a parallel algorithm developed by parallel programming to improve 3D NLGC IP inversion. First the theory of the NLGC algorithm for 3D IP inversion is presented. Second, implementation of the MPI and GPU parallel algorithm is briefly introduced. Finally, the efficiency of inversion codes is analyzed, with tables and synthetic data examples.

2. Forward

The electrical potential for arbitrary conductivity in a half-space is given by the differential equation:

$$\begin{aligned} \nabla \cdot [\boldsymbol{\sigma}(x, y, z) \nabla \mathbf{U}_\sigma(x, y, z)] \\ = -\mathbf{I} \delta(x - x_0) \delta(y - y_0) \delta(z - z_0), \end{aligned} \quad (1)$$

where $\boldsymbol{\sigma}$ is the conductivity of rocks (S/m); \mathbf{U}_σ denotes the potential (V) in absence of IP effect; \mathbf{I} is the electric current. The quantity on the right-hand side is in Ampere/meter³, while currents are in Ampere; and δ is the Dirac delta function. The Neumann boundary condition is applied for the earth-air interface; and a mixed boundary condition is adopted at infinite boundaries. The equation can be written as [10]

$$\begin{aligned} \frac{\partial \mathbf{U}}{\partial \mathbf{n}} &= 0, \\ \frac{\partial \mathbf{U}}{\partial \mathbf{n}} + \frac{\cos \theta}{r} \mathbf{U} &= 0, \end{aligned} \quad (2)$$

where θ is the angle between the radial distances from the source point to the outward normal spatial coordinate \mathbf{n} on the outer boundary. Then, the forward mapping operator \mathcal{F} is defined as

$$\mathbf{U}_\sigma = \mathcal{F}[\boldsymbol{\sigma}]. \quad (3)$$

If ground is chargeable, then the potential \mathbf{U}_η , which is induced by the application of a constant current will be different from \mathbf{U}_σ . According to Siegel's formulation [11], with the conductivity replaced by $\boldsymbol{\sigma}^* = \boldsymbol{\sigma}(1 - \boldsymbol{\eta})$, the effect of the chargeability of ground can be modeled by direct current resistivity forward mapping operator \mathcal{F} . Then, apparent chargeability $\boldsymbol{\eta}_a$ can be defined by equation

$$\boldsymbol{\eta}_a = \frac{\rho_a^* - \rho_a}{\rho_a^*}, \quad (4)$$

where ρ_a^* and ρ_a are the apparent resistivity computed by the potential \mathbf{U}_η and \mathbf{U}_σ , respectively. So the apparent chargeability can be computed by two DC resistivity modelings with conductivity $\boldsymbol{\sigma}^*$ and $\boldsymbol{\sigma}$.

2.1. Finite Difference Method. Usually, there are two ways to discretize the differential equation: the finite difference method and the finite element method. The finite difference method was applied for 3D forward modeling in the present study. The matrix form of (1) can be written as

$$\mathbf{A}\mathbf{U} = \mathbf{b}, \quad (5)$$

where \mathbf{A} is the coefficient matrix; \mathbf{U} is the potential vector; and \mathbf{b} is a vector containing the locations of the positive and negative current sources. There are many numerical methods for solving linear equations; of these, the ILU0 preconditioner and the BICGSTAB iterative method [12] was chosen to solve (5) in serial algorithm. The ILU0 preconditioner is an incomplete LU factorization with zero fill-in [13]:

$$\mathbf{L} \cdot \mathbf{U} \approx \mathbf{A}, \quad (6)$$

where \mathbf{L} is the lower triangular and \mathbf{U} is the upper triangular.

2.2. Forward Test. In order to test the accuracy of the forward algorithm, we compared our result of using the proposed BICGSTAB method with results from the preconditioned conjugate gradient (CGPC) algorithm [14]. The model in the present study consisted of a 10 Ω -m rectangular prism measuring 40 m \times 40 m \times 50 m embedded in a 100 Ω -m homogeneous half-space. The top of the model was located 30 m below ground surface. Four data acquisition methods were used for recording pole-pole data: surface-surface, surface-borehole, borehole-surface, and borehole-borehole (Figure 1). Their responses are shown in Figure 2, which indicates that the proposed BICGSTAB algorithm produced highly accurate results.

3. Inverse

Based on the result of resistivity result, nonlinear conjugate gradients IP inversion was adopted in this study since it constrains the model and is a nonlinear inversion method [15]. The objective function in the problem is in two parts:

$$\begin{aligned} \Phi &= (\boldsymbol{\eta}_{\text{obs}} - \boldsymbol{\eta}_{\text{ai}})^T \mathbf{C}_d^{-1} (\boldsymbol{\eta}_{\text{obs}} - \boldsymbol{\eta}_{\text{ai}}) \\ &+ \lambda (\boldsymbol{\eta}_i - \boldsymbol{\eta}_{\text{ref}})^T \mathbf{L}^T \mathbf{L} (\boldsymbol{\eta}_i - \boldsymbol{\eta}_{\text{ref}}), \end{aligned} \quad (7)$$

where $\boldsymbol{\eta}_{\text{obs}}$ is a vector of the observed data; $\boldsymbol{\eta}_{\text{ai}}$ is a vector of the calculated data obtained by the forward algorithm in the i th inversion iteration; \mathbf{C}_d is the random error matrix; λ is the Lagrange multiplier that balances the effect of data misfit and model regularization during the iteration; $\boldsymbol{\eta}_{\text{ref}}$ is the a priori model parameters vector; $\boldsymbol{\eta}$ is the model parameters vector at a given iteration; and \mathbf{L} is the second-difference Laplacian. The corresponding gradients of the objective function are expressed as

$$\mathbf{g} = -2\mathbf{J}^T \mathbf{C}_d^{-1} \mathbf{e} + 2\lambda \mathbf{L}^T \mathbf{L} (\boldsymbol{\eta}_i - \boldsymbol{\eta}_{\text{ref}}), \quad (8)$$

where \mathbf{J} is the Jacobian matrix and \mathbf{e} is the vector containing data residuals. The initial search direction $\mathbf{p}_0 = \mathbf{h}_0$ is obtained from

$$\mathbf{h} = \mathbf{C}\mathbf{g}, \quad (9)$$

where \mathbf{C} is the precondition factor, written as

$$\mathbf{C} = (\gamma \mathbf{I} + \lambda \mathbf{L}^T \mathbf{L})^{-1}, \quad (10)$$

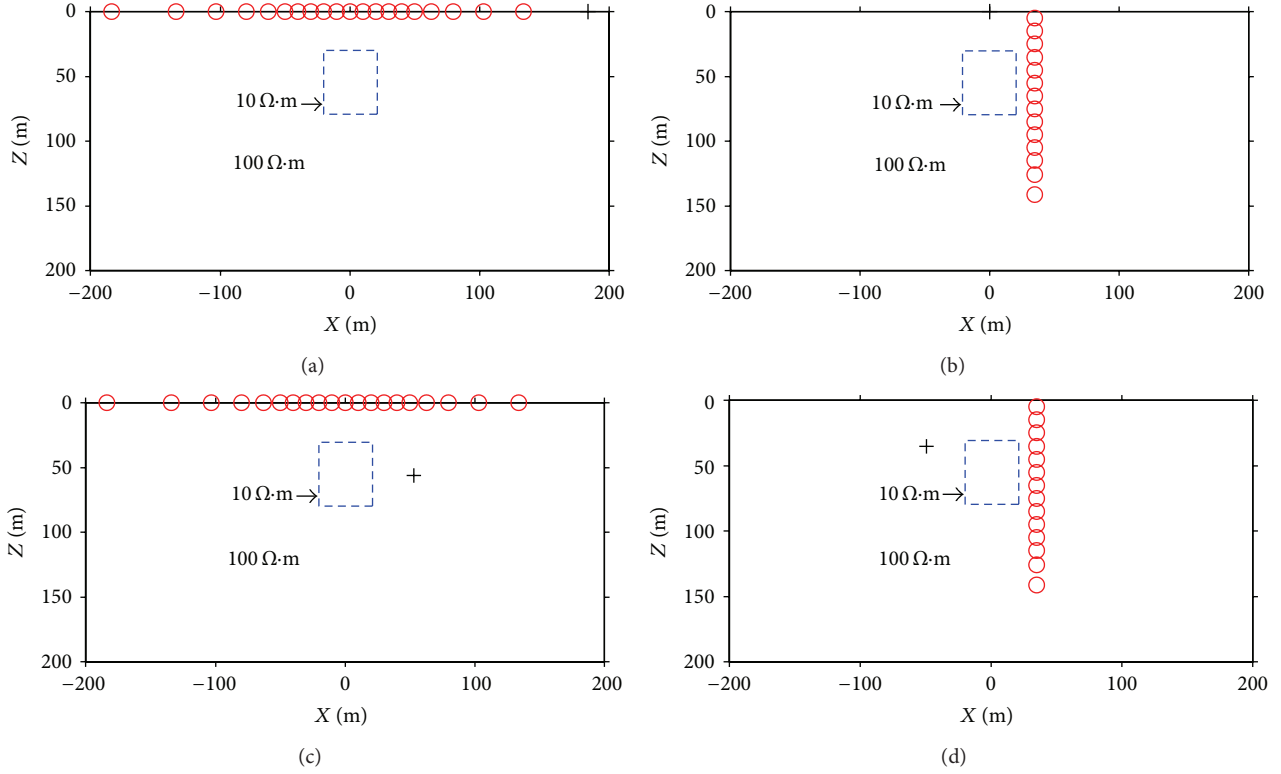


FIGURE 1: 3D model of high conductivity in the main profile ($y = 0$ m) (black crosses = current electrodes; red circles = receivers).

where γ is a specific scalar. The search direction in the i th inversion iteration can be defined:

$$\mathbf{p}_i = \begin{cases} -\mathbf{h}_i + \beta_i \mathbf{p}_{i-1}, & i = 1, 2, 3, \dots \\ -\mathbf{h}_i, & i = 0. \end{cases} \quad (11)$$

The updated step size is then obtained from

$$\mathbf{f} = \mathbf{J}\mathbf{p} \quad (12)$$

$$\alpha = -\frac{\mathbf{p}^T \mathbf{g}}{2(\mathbf{f}^T \mathbf{C}_d^{-1} \mathbf{f} + \lambda \mathbf{p}^T \mathbf{L}^T \mathbf{L} \mathbf{p})}.$$

The gradients and updated step are important parts of the inversion; $\mathbf{J}\mathbf{p}$ and $\mathbf{J}^T \mathbf{C}_d^{-1} \mathbf{e}$ are calculated directly as the characteristics of nonlinear conjugate gradients inversion. From (4) and (5), the sensitivity matrix \mathbf{J} can be derived as

$$\mathbf{J} = \frac{\partial \boldsymbol{\eta}_a}{\partial \boldsymbol{\eta}} = -\frac{\boldsymbol{\rho}_a}{(\boldsymbol{\rho}_a^*)^2} \cdot \frac{\partial \boldsymbol{\rho}_a^*}{\partial \boldsymbol{\sigma}^*} \cdot \boldsymbol{\sigma}. \quad (13)$$

4. MPI and GPU Parallel Programming

The main parallel computations in the NLCG inversion of 3D resistivity data are the forward and pseudo-forward modeling. Forward and pseudo-forward modeling of the same source is mainly done to form the coefficient matrix \mathbf{A} and the right-hand vector \mathbf{b} and solve the equation in a similar

fashion to (5). Because forward and pseudo-forward modeling is independent for each source, MPI parallel computation is suitable for this calculation.

We developed a parallel procedure based on the 3D NLCG serial procedure by calling Linux system interfaces with Intel Fortran compiler and used the principal-subordinate pattern for programming. The main process controls the assigning and sending tasks for every process and of reclaiming and assembling the results from other processes. The subordinate processes calculate the task sent from the main process and send the result back. 3D parallel inversion divides the 3D forward and pseudo-forward modeling tasks into several subordinate processes which are implemented in separate parallel execution nodes. The main process (NO. 0) reads all input parameters and broadcasts command messages to all subordinate processes. The subordinate processes finished calculation and sent result back to main process.

A workstation with a GPU card (NVIDIA Tesla C2075) is very fast for computation. Our parallel code was developed using CUDA (6.0) with the solver of CULA Sparse (S6) library. The BICGSTAB iteration method used in the serial code for solving linear equations was replaced by calling CULA Sparse library that provides iterative solvers for sparse systems. Therefore, the CULA iterative matrix solver is called during forward and pseudo-forward modeling in different processes. Table 1 shows the iteration times and runtimes of six different methods. These methods choose the same ILU0 preconditioner, double precision data type, and compressed sparse row (CSR) matrix storage format. "Iteration times"

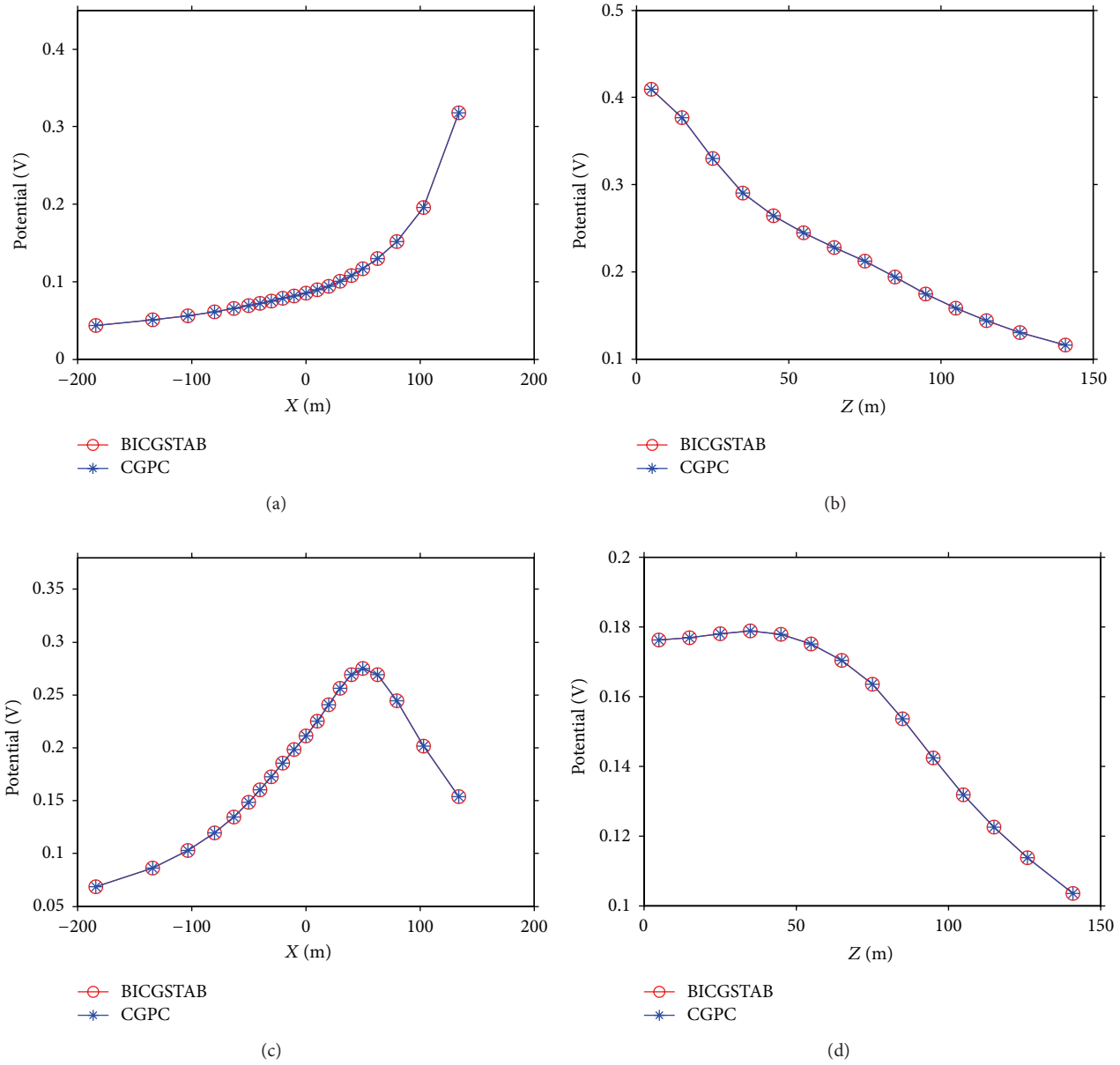


FIGURE 2: Forward modeling results from the BICGSTAB and CGPC numerical methods: (a) surface method (transmitters and receivers both on surface); (b) surface-borehole method (transmitters on the surface and receivers in borehole); (c) borehole-surface method (transmitters in borehole and receivers at surface); (d) borehole-borehole method (transmitters and receivers both in borehole).

TABLE 1: Iteration times and runtimes of different method for linear equation computation.

| Method | Iteration times | Overhead time (S) | Preconditioner time (S) | Solver time (S) | Total time (S) |
|-------------|-----------------|-------------------|-------------------------|-----------------|----------------|
| CG | 261 | 0.15 | 0.30 | 1.79 | 2.24 |
| BICG | 261 | 0.15 | 0.29 | 6.76 | 7.20 |
| BICGSTAB | 186 | 0.15 | 0.30 | 2.47 | 2.92 |
| BICGSTAB(L) | 87 | 0.15 | 0.29 | 2.25 | 2.69 |
| GMRES | 20 | 0.15 | 0.30 | 3.17 | 3.62 |
| MINRES | 245 | 0.15 | 0.29 | 1.87 | 2.30 |

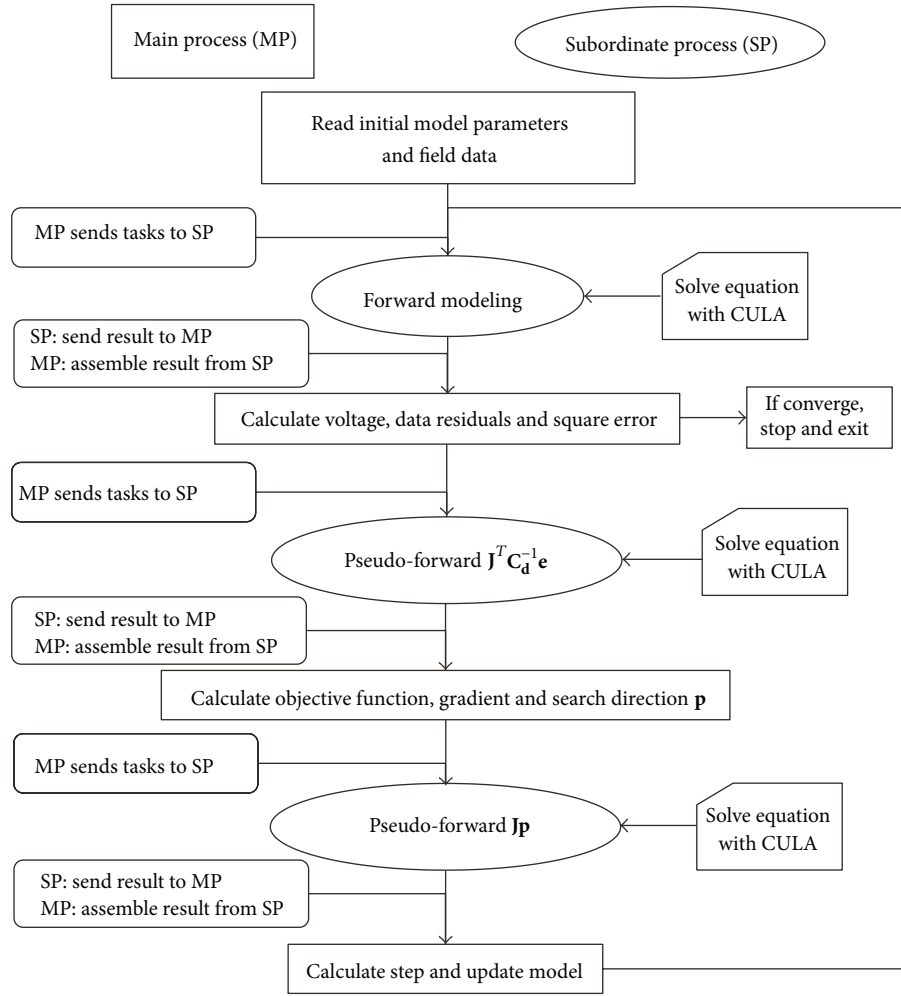


FIGURE 3: Flowchart of parallel algorithm of the 3D IP NLCG inversion. Rectangles and ovals indicate main process and subordinate process, respectively. The cards on the right-hand side of the ovals indicate that the iterative matrix solvers are called for forward and pseudo-forward modeling.

are the iteration times when achieving the convergence error 10^{-6} . “Overhead time” means the time that the memory transfers to GPU. “Preconditioner time” is the time for the preconditioner generative portion of the iterative solver. “Solver time” represents the time spent on iteration solver. “Total time” is the sum of overhead time, preconditioner time, and solver time. It is easy to know (from Table 1) that the CG method is the fastest one; the GMRES method spends the smallest number of iteration times. In our program, overall consideration, the CSR matrix storage format, ILU0 preconditioner, and CG iteration solver were chosen for solving linear equations.

The details of our parallel programming are shown in Figure 3. The flowchart of NLCG inversion shows that one forward modeling and two pseudo-forward modelings are conducted for each inversion iteration.

5. Numerical Experiments and Discussion

5.1. Examples with Synthetic Data. To reflect the influence of borehole data in the inversion, we carried out four field

scenarios for a synthetic simple resistivity model consisting of a $10 \Omega\text{-m}$ and 20% chargeability rectangular prism ($40 \text{ m} \times 40 \text{ m} \times 70 \text{ m}$) embedded in a $100 \Omega\text{-m}$ and 1% chargeability background. The top of the model is 30 m below ground surface (Figure 4). The input apparent chargeability with 5% Gaussian random noise was generated by applying 3D IP forward finite difference modeling algorithm for these four tests. The reference chargeability models were a homogeneous 1% half-space in the inversion. And all the four experiments were run using a fixed Lagrange multiplier $\lambda = 5000$ used to balance the effect of data misfit and model regularization. When one source starts working, all the receivers will record using pole-dipole.

For the first test, 189 current electrodes and 189 receiver electrodes were distributed equally along nine survey lines on the surface (Figure 5). Data was recorded using left- and right-side pole-dipole (AMN and MNB) in this surface resistivity method. The minimum distance between each consecutive electrode was 10 m and maximum distance was 50 m, giving a total of 33,660 data points for the entire survey.

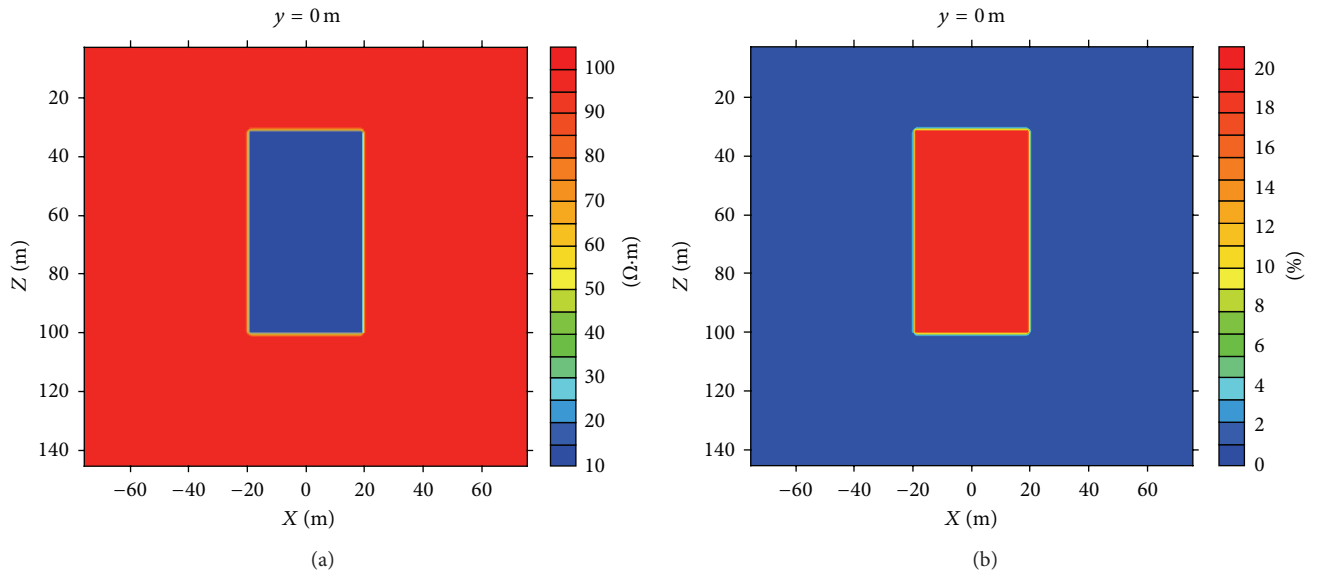


FIGURE 4: Synthetic resistivity and chargeability model in the main profile ($y = 0$ m); (a) shows resistivity model; (b) shows chargeability model.

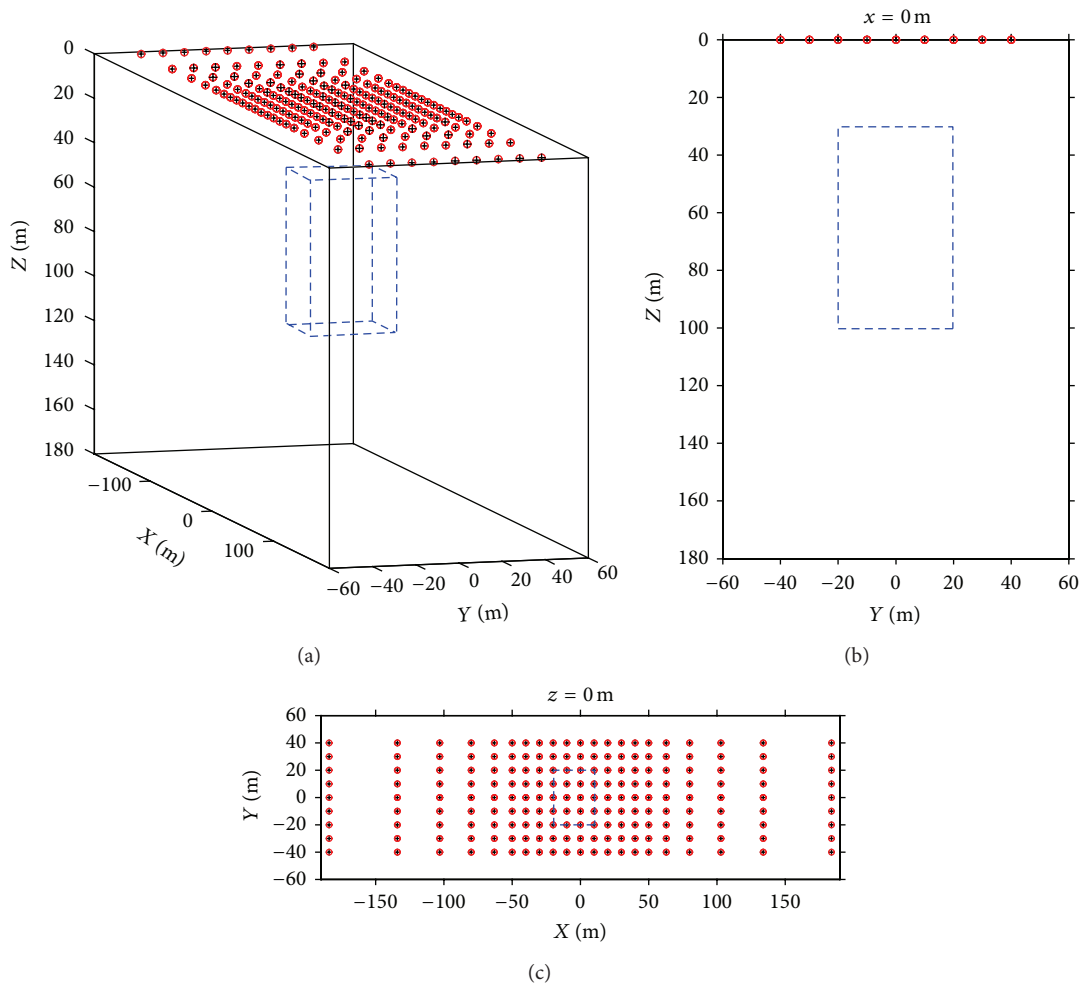


FIGURE 5: Observation system for surface resistivity method; (a) shows 3D graph of acquisition mode; (b) shows synthetic model and electrodes in the Y direction main profile; (c) shows surface electrodes and projection of anomalous body on the ground (black crosses = current electrodes; red circles = receivers).

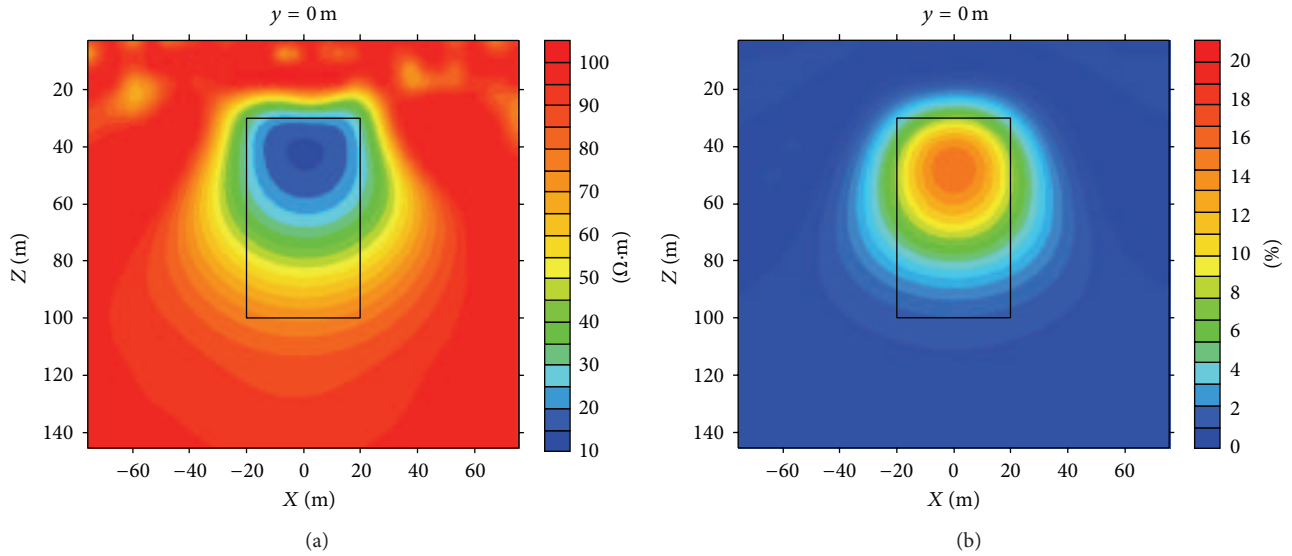


FIGURE 6: Inversion results for surface method in the main profile ($y = 0$ m); (a) shows corresponding resistivity inversion results; (b) shows corresponding inversion chargeability results.

TABLE 2: Statistical runtimes for linear equation computation.

| Processor | Method | Number of processes | 3D grid size | Number of sources in each process | Runtime (seconds) | Number of iterations |
|-----------|----------|---------------------|--------------------------|-----------------------------------|-------------------|----------------------|
| CPU | BICGSTAB | 1 | $89 \times 89 \times 87$ | 1 | 23.028 | 400 |
| GPU | BICGSTAB | 1 | $89 \times 89 \times 87$ | 1 | 6.881 | 400 |

The vertical sections of resistivity and chargeability inversion result are illustrated in Figure 6 from Y direction. Both the resistivity and chargeability anomalies are located near the top of the block, not at its center, and the target values of anomalies are not recovered. Furthermore, the boundary of the anomalous body is not delineated very well, especially in the bottom of block.

In the second test, the first test was repeated, but with the original distribution of electrodes and receivers enhanced by 28 pairs of additional receivers placed at similar positions in two boreholes. Thus, for each source, 28 additional independent voltage measurements were taken from two boreholes, combining the surface and surface-borehole methods. The minimum distance between borehole electrodes was 10 m, and maximum distance was 50 m (Figure 7). The total number of data points was 38,952. Figure 8 shows the vertical sections of resistivity and chargeability inversion result. Although the anomaly is not located in the center of block, it is closer than in the first test. And the boundary of the anomalous body is delineated more clearly than the result of first test. Overall, the top of block is recovered better; because of the data from borehole, the anomaly is stretched in the Z -direction.

The third experiment combined the borehole-surface and borehole-borehole methods. The distributions of receivers were the same as in the second test, and 28 current electrodes

were put in similar positions into two boreholes (Figure 9) and total 5,824 sets of voltage data were joined to the inversion calculation. Figure 10 shows improved inversion results compared to those from the first and second tests: the anomalous body appears at the center of the block, and its boundary is clearly delineated; furthermore, the bottoms of anomalies are relatively wider.

In the fourth and final test, all the receivers and current electrodes from test one to test three were used (Figure 11), combining the surface, surface-borehole, borehole-surface, and borehole-borehole methods. The entire experiment yielded 44,776 sets of synthetic data. The inversion captured the correct location of the block in the vertical sections and produced the best result of the four tests. Compared with second and third test, the boundary is more clearly seen in the vertical section (Figure 12). The magnitudes of the resistivity and chargeability values and the position of the anomaly matched the theoretical model reasonably well.

5.2. Computation Efficiency. In order to test the GPU computation efficiency, the runtime of linear equation computation with GPU was compared with CPUs in the same method, processes, sources, and iteration times (see Table 2). Next, we tested the computation efficiency of the parallel algorithm of NLCG. Some four-way tests were carried out in two modes (MPI and MPI + GPU): (i) a parallel algorithm test using

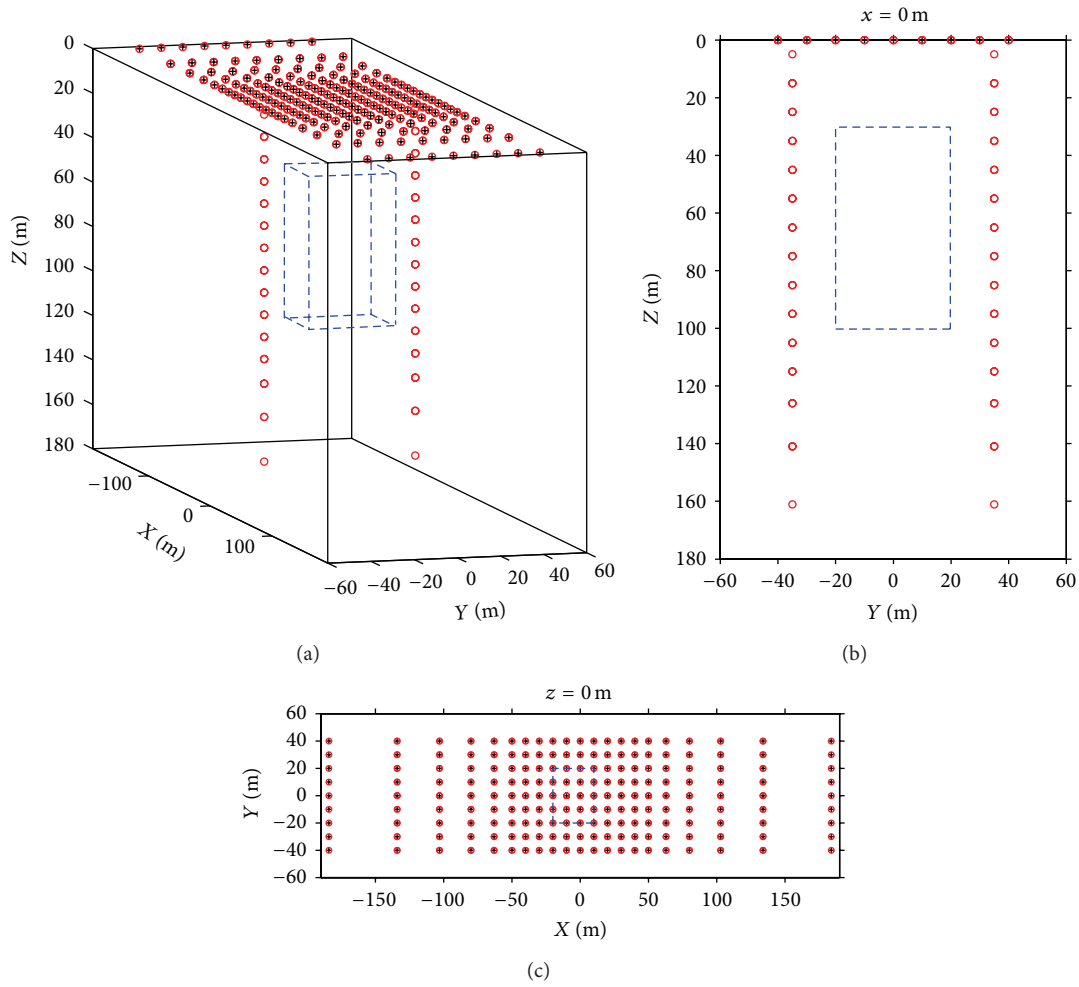


FIGURE 7: Observation system for surface-borehole and surface method; (a) shows 3D graph of acquisition mode; (b) shows synthetic model and electrodes in the Y direction main profile; (c) shows surface electrodes and projection of anomalous body on the ground (black crosses = current electrodes; red circles = receivers).

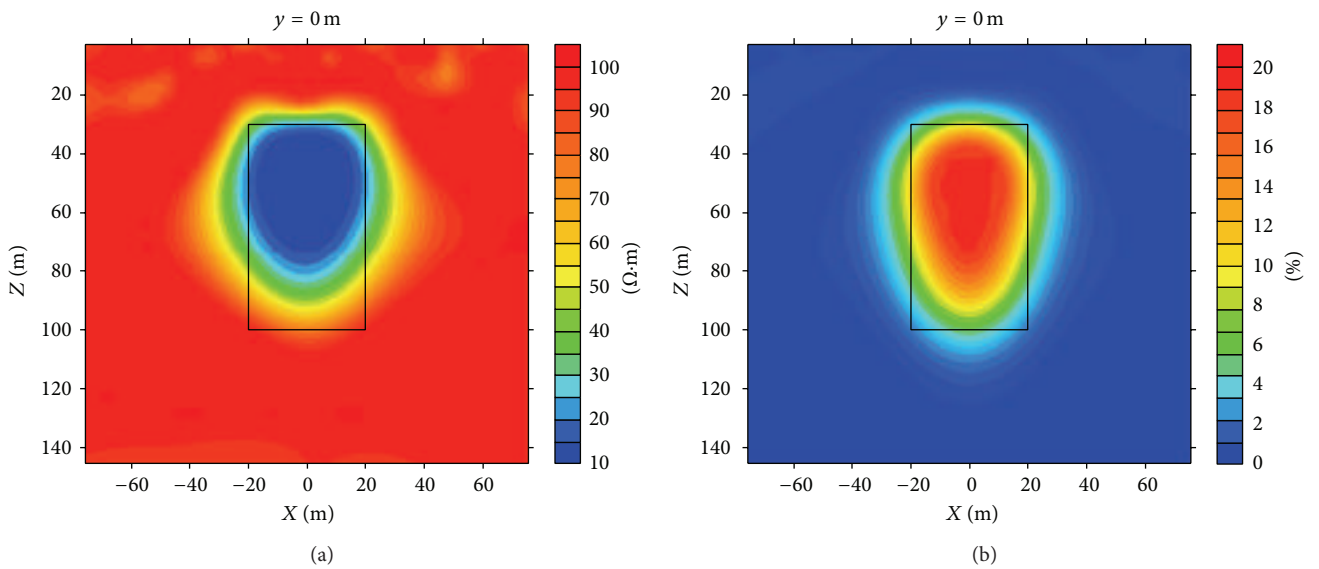


FIGURE 8: Inversion results for surface-borehole and surface method in the main profile ($y = 0$ m); (a) shows corresponding resistivity inversion results; (b) shows corresponding inversion chargeability results.

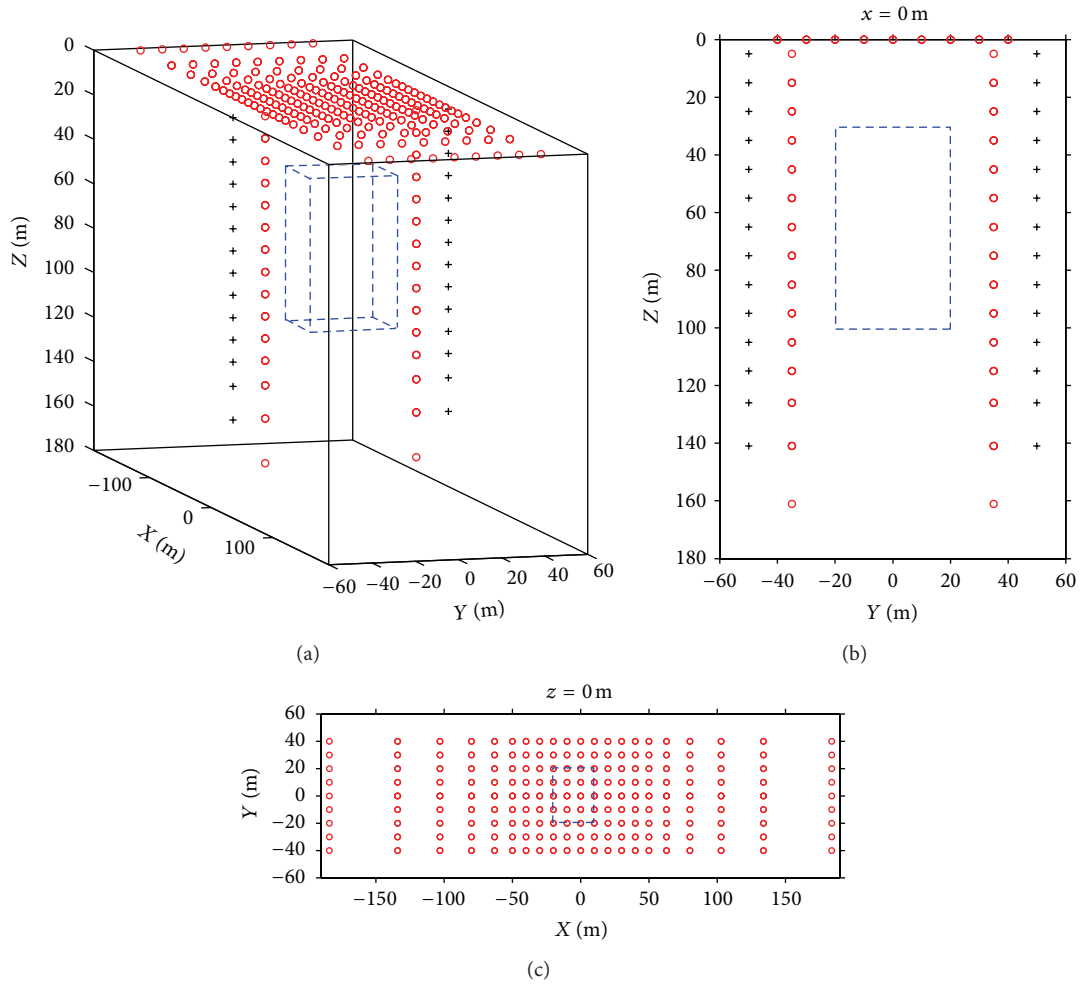


FIGURE 9: Observation system for combined borehole-surface and borehole-borehole method; (a) shows 3D graph of acquisition mode; (b) shows synthetic model and electrodes in the Y direction main profile; (c) shows surface electrodes and projection of anomalous body on the ground (black crosses = current electrodes; red circles = receivers).

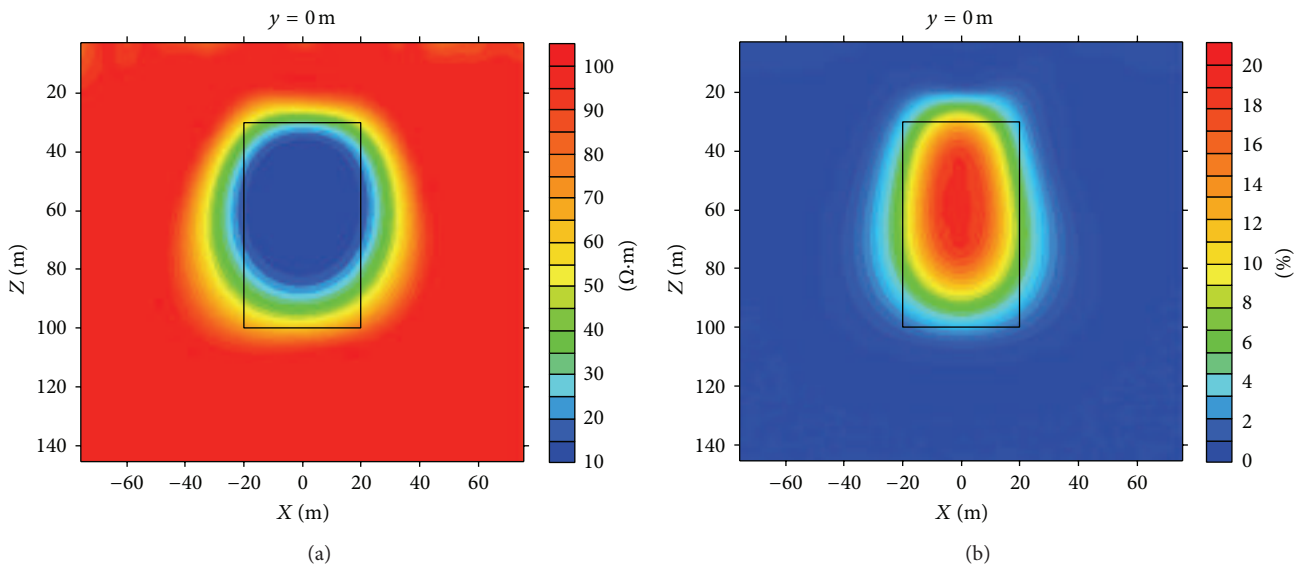


FIGURE 10: Inversion results for combined borehole-surface and borehole-borehole method in the main profile ($y = 0$ m); (a) shows corresponding resistivity inversion results; (b) shows corresponding inversion chargeability results.

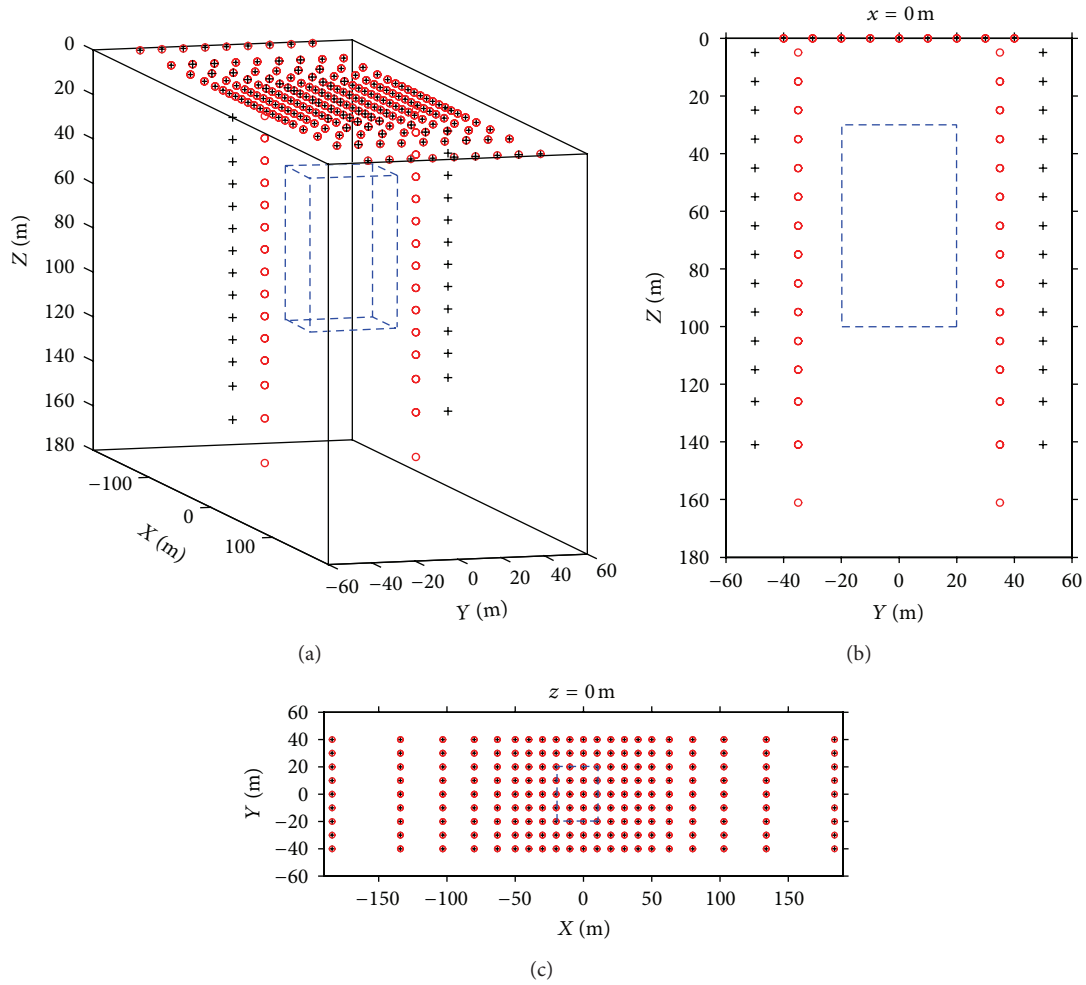


FIGURE 11: Observation system for combined surface, surface-borehole, borehole-surface, and borehole-borehole method; (a) shows 3D graph of acquisition mode; (b) shows synthetic model and electrodes in the Y direction main profile; (c) shows surface electrodes and projection of anomalous body on the ground (black crosses = current electrodes; red circles = receivers).

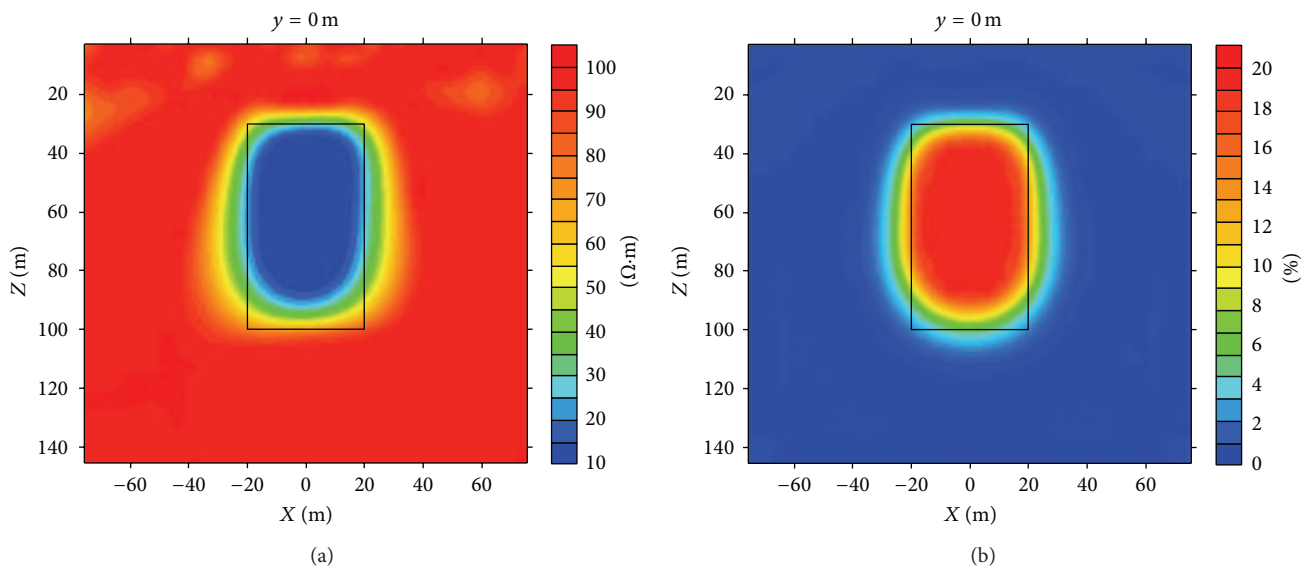


FIGURE 12: Inversion results for combined surface, surface-borehole, borehole-surface, and borehole-borehole method in the main profile ($y = 0$ m); (a) shows corresponding resistivity inversion results; (b) shows corresponding inversion chargeability results.

TABLE 3: Statistical runtimes for the prism model inversion.

| Mode | Number of computers | Number of sources in each process | Runtime (hours) | Percentage of serial algorithm (%) |
|-----------|---------------------|-----------------------------------|-----------------|------------------------------------|
| MPI | 1 | 60 | 4.643 | 100% |
| | 2 | 30 | 2.414 | 52% |
| | 3 | 20 | 1.625 | 35% |
| | 4 | 15 | 1.207 | 26% |
| GPU + MPI | 1 | 60 | 1.254 | 27% |
| | 2 | 30 | 0.651 | 14% |
| | 3 | 20 | 0.419 | 9% |
| | 4 | 15 | 0.327 | 7% |

a single machine with one process; (ii) a parallel algorithm test using two machines with one process in each; (iii) a parallel algorithm test using three machines with one process in each; and (iv) a parallel algorithm test using four machines with one process in each. All the four tests with the grid size of $67 \times 67 \times 65$ were computed by ten inversion iteration times. Table 3 shows the statistical runtimes for the prism model inversion for the four tests. “Runtime (hours)” is the time spent until all the processes stop. “Percentage of serial algorithm (%)” is the time spent by the parallel algorithm as a percentage of the time spent by the serial algorithm.

In Table 2, the runtime with GPU is about 3.8 times the runtime with CPU. Table 3 shows that the efficiency of the algorithm is greatly enhanced by MPI + GPU parallel computation, especially when using four machines. The runtime using MPI + GPU is about 3.8 times the runtime using MPI alone, with the same number of machines and processes.

6. Conclusion

Based on the analysis of NLCG algorithm for 3D IP inversion with MPI and GPU parallel programming, we developed a parallel NLCG 3D resistivity inversion code for data generated by surface, borehole-surface, surface-borehole, and borehole-borehole IP methods. The inversion results show that data from the borehole improves the quality of the inversion and delineates the boundary of an anomalous body more clearly. The borehole data can constrain the bottom of the anomalous body very well; similarly, the surface data can constrain the top of block. We tested the parallel code using synthetic data and compared it with a serial procedure. This showed that our proposed parallel algorithm not only is effective but also greatly improved the speed of inversion. It also shows that the computing speed of a workstation with a GPU card (NVIDIA Tesla C2075) is faster than one with CPU cores. The tables demonstrate that the CG method of CULA library is suitable for double precision data type and symmetric positive definite coefficient matrix; the equation solved by the iterative method using GPU is faster than that with CPU; the parallel algorithm accelerates

inversion in cases when high efficiency is required; furthermore, the use of larger numbers of machines enhances this effect.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

Acknowledgments

The authors would like to thank the support by National Natural Science Foundation of China (no. 41374078) and the Ministry of Land and Resources of Geological Survey Projects (nos. 12120113086100 and 12120113101300).

References

- [1] M. Dabas, A. Tabbagh, and J. Tabbagh, “3-D inversion in subsurface electrical surveying—I. Theory,” *Geophysical Journal International*, vol. 119, no. 3, pp. 975–990, 1994.
- [2] W. R. Petrick Jr., W. R. Will, and S. H. Ward, “Three-dimensional resistivity inversion using alpha centers,” *Geophysics*, vol. 46, no. 8, pp. 1148–1162, 1981.
- [3] Y. Li and D. W. Oldenburg, “Inversion of 3-D DC resistivity data using an approximate inverse mapping,” *Geophysical Journal International*, vol. 116, no. 3, pp. 527–537, 1994.
- [4] D. W. Oldenburg and Y. Li, “Inversion of induced polarization data,” *Geophysics*, vol. 59, no. 9, pp. 1327–1341, 1994.
- [5] X. P. Wu, “Rapid 3-D inversion of induced polarization data using conjugate gradient method,” *Coal Geology & Exploration*, vol. 23, no. 5, pp. 62–64, 2004 (Chinese).
- [6] P. S. Pacheco and Q. N. Deng, *An Introduction to Parallel Programming*, China Machine Press, Beijing, China, 2013 (Chinese).
- [7] Z. X. Chen, X. H. Meng, G. F. Liu, and L. H. Guo, “The GPU-based parallel calculation of gravity and magnetic anomalies for 3D arbitrary bodies,” *Geophysical & Geochemical Exploration*, vol. 36, no. 1, pp. 117–121, 2012 (Chinese).
- [8] Z. X. Chen, X. H. Meng, L. H. Guo, and G. F. Liu, “Three-dimensional fast forward modeling and the inversion strategy for large scale gravity and gravimetry data based on GPU,” *Chinese Journal of Geophysics*, vol. 55, no. 12, pp. 4069–4077, 2012 (Chinese).
- [9] R. Farber, *CUDA Application Design and Development*, China Machine Press, Beijing, China, 1st edition, 2013, (Chinese).
- [10] A. Dey and H. F. Morrison, “Resistivity modelling for arbitrarily shaped two-dimensional structures,” *Geophysical Prospecting*, vol. 27, no. 1, pp. 106–136, 1979.
- [11] H. O. Seigel, “Mathematical formulation and type curves for induced polarization,” *Geophysics*, vol. 24, no. 3, pp. 547–565, 1959.
- [12] H. A. van der Vorst, “Bi-CGSTAB: a fast and smoothly converging variant of Bi-CG for the solution of nonsymmetric linear systems,” *SIAM Journal on Scientific and Statistical Computing*, vol. 12, pp. 631–644, 1992.
- [13] Y. Saad, “ILUT: a dual threshold incomplete ILU factorization,” *Numerical Linear Algebra with Applications*, vol. 1, no. 4, pp. 387–402, 1994.

- [14] K. Spitzer, "A 3-D finite-difference algorithm for DC resistivity modelling using conjugate gradient methods," *Geophysical Journal International*, vol. 123, no. 3, pp. 903–914, 1995.
- [15] W. Rodi and R. L. Mackie, "Nonlinear conjugate gradients algorithm for 2-D magnetotelluric inversion," *Geophysics*, vol. 66, no. 1, pp. 174–187, 2001.



Hindawi

Submit your manuscripts at
<http://www.hindawi.com>

