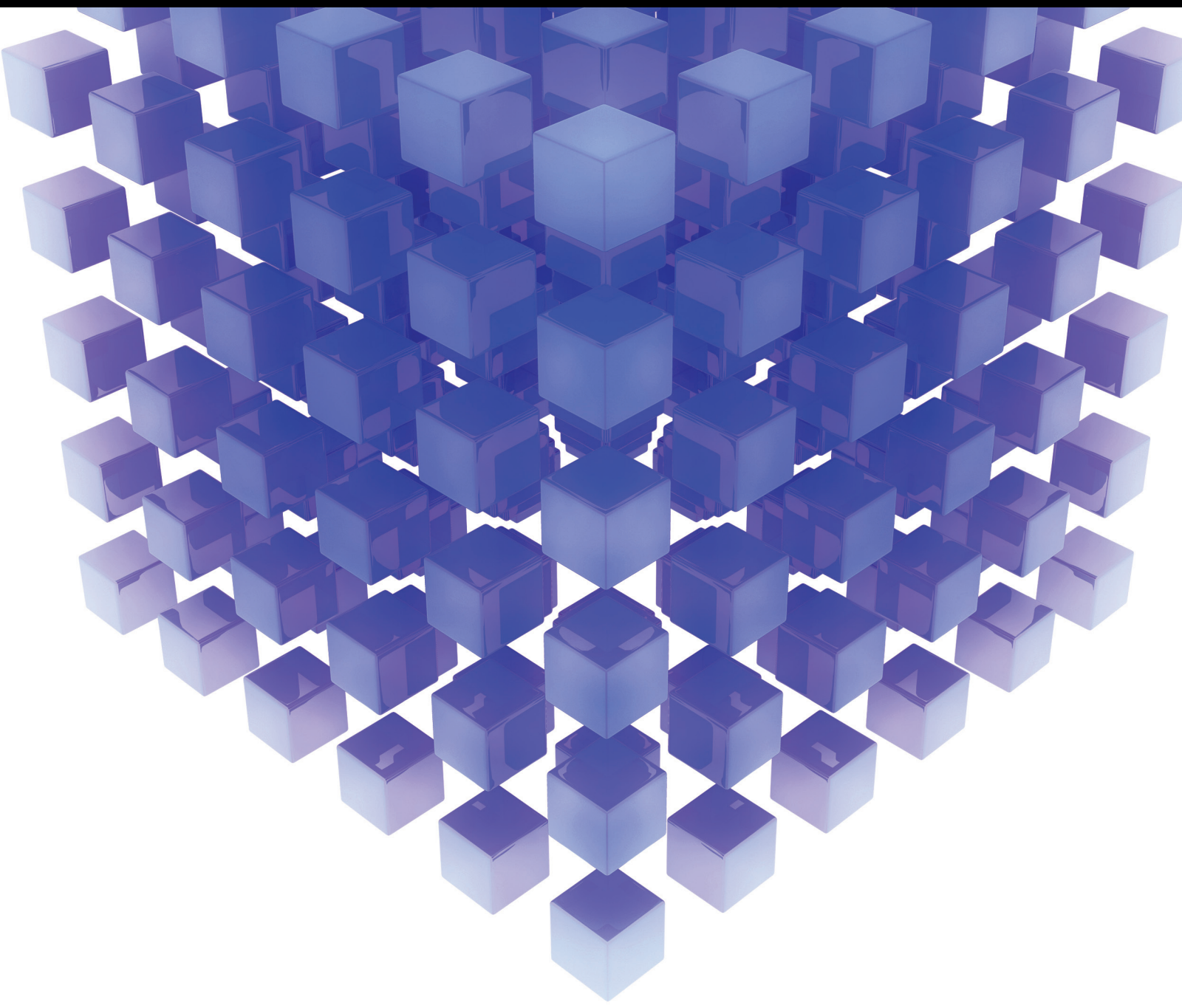
Mathematical Problems in Engineering

Recent Theory and Applications on Inverse Problems 2014

Guest Editors: Fatih Yaman, Valery G. Yakhno, Caner Özdemir, Tzu-Yang Yu, and Roland Potthast



Recent Theory and Applications on Inverse Problems 2014

Mathematical Problems in Engineering

Recent Theory and Applications on Inverse Problems 2014

Guest Editors: Fatih Yaman, Valery G. Yakhno, Caner Özdemir, Tzu-Yang Yu, and Roland Potthast

Editorial Board

Contents

Hindawi Publishing Corporation
Mathematical Problems in Engineering
Volume 2015, Article ID 403729, 2 pages
http://dx.doi.org/10.1155/2015/403729

Editorial
Recent Theory and Applications on Inverse Problems 2014

Fatih Yaman,[1] Valery G. Yakhno,[2] Caner Özdemir,[3] Tzu-Yang Yu,[4] and Roland Potthast[5,6]

[1]*Department of Electrical and Electronics Engineering, Izmir Institute of Technology, Gülbahce-Urla, 35430 Izmir, Turkey*
[2]*Department of Electrical and Electronics Engineering, Dokuz Eylul University, 35160 Izmir, Turkey*
[3]*Department of Electrical and Electronics Engineering, Mersin University, 33343 Mersin, Turkey*
[4]*Department of Civil and Environmental Engineering, University of Massachusetts Lowell, Lowell, MA 01854-2827, USA*
[5]*Department of Mathematics, University of Reading, Whiteknights, P.O. Box 220, Reading RG6 6AX, UK*
[6]*German Meteorological Service, Deutscher Wetterdienst Research and Development, Head Division FE 12 (Data Assimilation), Frankfurter Strasse 135, 63067 Offenbach, Germany*

Correspondence should be addressed to Fatih Yaman; fatihyaman@iyte.edu.tr

Received 25 December 2014; Accepted 25 December 2014

This special issue presents some very interesting recent developments in the area of inverse problems. The scope of the issue covers wide range of disciplines, for example, electromagnetics, acoustics, heat conduction, and image processing, from theory and application point of view. In the following, we give very brief descriptions of the published papers.

M. R. Santos et al. propose the estimation of heat flux at the chip-tool interface using inverse technique. The paper demonstrates an elaborate experimentation. The nonlinear heat diffusion equation is solved via a 3D numerical code and the temperature distribution is predicted using finite volume elements. For the inverse problem, the function specification method is employed. Heat fluxes at the tool-workpiece interface are estimated using inverse problems methods and experimental temperatures. The results are matched in a fair way with those of the tool-work thermocouple technique for cutting parameters on cutting edge temperature.

C. He et al. present an adaptive TGV-based model for noise removal. The aim of the study is to achieve a balance between edge preservation and region smoothness for image denoising. The variable splitting and the classical augmented Lagrangian method (ALM) are employed in the solution of the proposed method. The authors observe that the proposed algorithm is effective in suppressing staircasing effect and preserving edges in images, and it is superior to some other famed adaptive denoising methods both in quantitative and in qualitative assessment.

X. Zeng proposes a bound alternative direction method which can be considered as an extension of alternating direction method of multipliers for the solution of ℓ_p $(p \in (0,1))$ minimization problems in image deblurring. The author reports that the experiments on a set of image deblurring problems have shown that the proposed method for the synthesis ℓ_p formulation is favorably competitive with the state-of-the-art algorithms for the synthesis ℓ_1 formulation.

C. Özdemir et al. present a review paper whose aim is to evaluate and compare the migration algorithms over different focusing methods such that the reader can decide which algorithm to use for a particular application of GPR. In the paper, the brief formulation and the algorithm steps for the hyperbolic summation, the Kirchhoff migration, the back-projection focusing, the phase-shift migration, and the $(\omega-k)$ migration are presented and the simulated and the measured examples that are used for the performance comparison of the presented algorithms are provided.

Y. He et al. investigate the flight maneuvering trajectories evaluation problem. The algorithm differentiation is used to realize the inverse computation of the rotor-flying robot system. The advantages of the proposed algorithms are mentioned as only desired positions and their derivatives and the feasibility of the trajectories (including inner states and inputs) can be evaluated and additionally an accurate pointwise numerical feasibility analysis of planned trajectories becomes possible.

C. Deng et al. consider a compound regularization based compressive sensing MRI reconstruction model, which exploits the NLTV regularization and wavelet approximate sparsity prior. For the algorithm, the variable splitting and augmented Lagrangian algorithm is applied to solve the compound regularization minimization problem. Authors report that the experiments on test images demonstrate that the proposed method leads to high SNR measure and more importantly preserves the details and edges of MR images.

J. Wang et al. propose a strategy using joint angle-Doppler representation basis which can determine a sparse target scenario in spatial domain at the same range for active radar in the forward-looking direction. According to the authors, the presented approach settles the trouble that traditional SAR and DBS techniques cannot provide an image for active radar in the line of sight and needs only single-receiver channel without any modification on traditional radar hardware. The given strategy shows good performance with different setup about SNR level and target numbers.

A. Palafox et al. pose an acoustic inverse scattering problem in a Bayesian inference perspective and simulate the posterior distribution using Markov chain Monte Carlo (MCMC). For the corresponding direct problem, the classical layer potential approach is employed and the problem was solved in a fast and reliable manner with parallel computing. The authors implemented the effective dimension method via Bayesian model selection where the normalizing constant for each model is approximated using the MCMC output for giving a parametric representation of the solution of the inverse problem.

D. Wang et al. apply the adjoint tidal model based on the theory of inverse problem to investigate the effect of bottom friction coefficient on the tidal simulations. In their work, the Bohai, Yellow, and East China Sea are simulated by assimilating altimeter data. Authors report that the simulated results with new empirical formulas are better than traditional schemes, such as the constant, different constant in different subdomain, and depth-dependent form.

F. Mohebbi and M. Sellier study an optimal shape design problem in aerodynamics. The aim of the inverse problem is finding the optimal shape of an airfoil placed in a potential flow at a given angle of attack should have such that the pressure distribution on its surface matches a desired one. To achieve this aim, a numerical method is investigated to generate a mesh over the airfoil surface and to solve for the flow equation. The authors report in their paper that the proposed sensitivity analysis method reduces the computation cost even for large number of the design variables and confirm accuracy and efficiency of the proposed shape optimization algorithm.

K. T. Iskakov and Zh. O. Oralbekova consider the direct and inverse problems for the geoelectric equations. They have showed that the considered problem is reducible to a system of the Volterra integral equations. A stability estimate has been proved for the solution of this system. As a result, a conditional stability estimate has been obtained for the solution of the considered inverse problem. The application of this conditional stability estimate is discussed.

H.-D. Yeh et al. propose an approach to solve complicated source release problems in groundwater contaminant plumes, containing unknowns of three location coordinates and several irregular release periods and concentrations. The authors employ an ordinal optimization algorithm (OOA), roulette wheel approach, and a source identification algorithm, based on simulated annealing, tabu search, and a three-dimensional groundwater flow and solute transport model. It is demonstrated that the proposed approach works effectively in the problems with different initial guesses of source location and measurement errors and with large suspicious areas and several source release periods and concentrations.

C. Ak and A. Yildiz propose an inverse approach to obtain a relation between applied voltage and displacement of the midpoint of fixed-fixed beam actuator. An inversely designed model and a modified formula which establishes a good approximation of the system with an unsophisticated representation are presented. In this direction, one can calculate required voltage for pull-in limit in a simpler way. On the other hand, the approach, for example, does not take the fringing effect into account for the sake of simplicity. However, employment of artificial optimization techniques is suggested for the improvement of the modified formula.

G. Jin et al. study the inversion of vertical eddy viscosity coefficient (VEVC) with a method based on an isopycnic coordinate internal tidal model in their paper. Numerical experiments are provided to examine the influence factors on the inversion of VEVCs in four aspects: independent point schemes (IPS), topography, the spatial distribution of VEVC, and the optimization methods. The authors found out in their investigations that the limited-memory Broyden–Fletcher–Goldfarb–Shanno (L-BFGS) method is a more effective method than the gradient descent method (GDM-S) in terms of the inversion of the VEVC. However, it is noted that the GDM-S is more convenient and controllable, so it should not be ignored and should be taken seriously as a choice for the inversion of the VEVC with spatial distribution.

Z. Ruan et al. study a backward problem for a time-fractional diffusion equation. This ill-posed problem is formulated as a regularized optimization problem. The authors propose a direct numerical algorithm for solving the stated problem. This algorithm is based on the adoption Tikhonov regularization to overcome ill-posedness. The regularization parameters are selected from computational experiments by the discrepancy principle. The numerical results confirm the robustness of the algorithm.

Acknowledgments

The guest editors would like to deeply thank all the authors, the reviewers, and the editorial board involved in the preparation of this issue.

Fatih Yaman
Valery G. Yakhno
Caner Özdemir
Tzu-Yang Yu
Roland Potthast

Hindawi Publishing Corporation
Mathematical Problems in Engineering
Volume 2015, Article ID 915793, 14 pages
http://dx.doi.org/10.1155/2015/915793

Research Article

Inversion Study of Vertical Eddy Viscosity Coefficient Based on an Internal Tidal Model with the Adjoint Method

Guangzhen Jin,[1] Qiang Liu,[2] and Xianqing Lv[1]

[1]*Laboratory of Physical Oceanography, Ocean University of China, Qingdao 266100, China*
[2]*College of Engineering, Ocean University of China, Qingdao 266100, China*

Correspondence should be addressed to Xianqing Lv; xqinglv@ouc.edu.cn

Received 28 March 2014; Revised 17 August 2014; Accepted 18 August 2014

Academic Editor: Fatih Yaman

Based on an isopycnic-coordinate internal tidal model with the adjoint method, the inversion of spatially varying vertical eddy viscosity coefficient (VEVC) is studied in two groups of numerical experiments. In Group One, the influences of independent point schemes (IPSs) exerting on parameter inversion are discussed. Results demonstrate that the VEVCs can be inverted successfully with IPSs and the model has the best performance with the optimal IPSs. Using the optimal IPSs obtained in Group One, the inversions of VEVCs on two different Gaussian bottom topographies are carried out in Group Two. In addition, performances of two optimization methods of which one is the limited-memory Broyden-Fletcher-Goldfarb-Shanno (L-BFGS) method and the other is a simplified gradient descent method (GDM-S) are also investigated. Results of the experiments indicate that this adjoint model is capable to invert the VEVC with spatially distribution, no matter which optimization method is taken. The L-BFGS method has a better performance in terms of the convergence rate and the inversion results. In general, the L-BFGS method is a more effective and efficient optimization method than the GDM-S.

1. Introduction

Internal tide, which is the internal wave of tidal frequency, is a ubiquitous phenomenon in the oceans. Rattray [1], Baines [2], Bell [3], Baines [4], Craig [5], Gerkema [6], and Llewellyn Smith and Young [7] have developed theoretical models and obtained some analytical solutions of internal tide on ideal topographies. These theoretical models helped them investigate the generation and propagation of internal tide. Although great progress has been made on the internal tide theory and some analytical works were carried out, only a small amount of solutions can be provided, due to the complexity of the problems. For this reason, quantitative analysis with practical significance still has to rely on the combination of numerical simulation, theoretical analysis, experiment, and observation. Numerical simulation is an effective method in marine research and has been widely used in the internal tide research. Kang et al. [8] investigated the M_2 internal tide near Hawaii with a two-dimensional, two-layered numerical model and confirmed that the internal tide was generated by barotropic forcing at the Hawaiian Ridge and propagated in north-northeast and south-southwest directions. Based on a high-precision three-dimensional Princeton Ocean Model (POM), Niwa and Hibiya [9] obtained the distribution of the M_2 internal tide in the Pacific Ocean using the TOPEX/Poseidon (T/P) satellite data. Cummins et al. [10] simulated the generation and propagation of internal tides near the Aleutian Ridge using T/P altimeter data. The comparison between the altimeter data and their model results showed good agreement for the phase, which also provided evidence for wave fraction near the Aleutian Ridge. With a three-dimensional POM, Niwa and Hibiya [11] investigated the distribution and the energy of the M_2 internal tide around the continental shelf edge in the East China Sea. Their numerical experiment results indicated that M_2 internal tides are effectively generated over prominent topographies such as sea ridges, island chains, and straits. Jan et al. [12] modified the POM to study the generation of the K_1 internal tide and its influence on surface tide in the South China Sea. The conversion from

the barotropic energy to the baroclinic energy over topographic ridges in the Luzon Strait was also estimated.

Determination of the vertical eddy viscosity coefficient (VEVC), which describes the vertical mixing in the ocean, plays an important role in the study of energy exchange and material transportation. The VEVC is regularly regarded as a constant in numerical models. Schemes to determine the VEVC mainly include the Prandtl mixing-length hypothesis model, the k-ε model, the Pacanowski-Philander mixing model [13], and some turbulent closure models that are more complicated. Many studies have been carried out to investigate the variation of the VEVC [14–18]. All these mentioned studies indicate that due to different intensions of the vertical mixing in sea water, the VEVC should not be treated as a constant but a parameter with spatial distribution.

Satellite remote sensing technology and other related technologies provide us with a large number of data. Thus, it is one of the most important missions in physical oceanography to make use of the data efficiently and precisely as well as to combine the observation data with present numerical models. Indeed, data assimilation with the adjoint method provides an effective access to these missions. The use of the adjoint method in marine science can be traced back to 1980s. The adjoint model is capable of optimizing control parameters in numerical simulation. Bennett and McIntosh [19] applied the weak constraint thought to solve the tidal problem and the geostrophic-flow problem. Yu and O'Brien [20] assimilated both meterological and oceanographic data into an oceanic Ekman layer model and deduced the unknown boundary condition, the unknown vertical eddy viscosity, and the current field. Based on a tidal model with a two-level leapfrog method, Lardner [21] inverted the open boundary conditions in three-test problems. Seiler [22] used the adjoint method to assimilate observations into a quasi-geostrophic ocean model and estimated the lateral boundary values in ideal experiments. Navon [23] wrote a summary of the parameter estimation in meteorology and oceanography in the view of applications with four- dimensional variational data assimilation techniques. Using an automatic differentiation compiler, Ayoub [24] constructed the adjoint model of the Massachusetts Institute of Technology Ocean General Circulation Model and inverted the open boundary conditions in the North Atlantic. Zhang and Lu [25] developed a three-dimensional nonlinear numerical tidal model with the adjoint method and designed several numerical experiments to estimate three kinds of parameters including the open boundary conditions, the bottom friction coefficients, and the vertical eddy viscosity coefficients. Zhang and Lu [26] employed a two-dimensional tidal model to study the inversion of the bottom friction coefficients in the Bohai Sea and the Yellow Sea with the adjoint method. Chen et al. [27] constructed a three-dimensional internal tidal model with the adjoint method and estimated six different kinds of open boundary conditions on fourteen types of topography. Based on a tidal model, Zhang and Chen [28] carried out several semiidealized experiments to estimate the partly and fully spatial varying open boundary conditions. Cao et al. [29] investigated the inversion of open boundary conditions with a three-dimensional internal tidal model and simulated the M_2 internal tide around Hawaii by assimilating T/P data.

There are two main objectives of this paper. One is to study the inversion of the VEVC with an internal tidal model and the adjoint method. According to the introductions above, a lot of studies have been carried out to investigate the inversion of the control parameters of internal tide such as the open boundary condition [29, 30] and the bottom friction condition [31, 32]. However, few works are found to study the inversion of VEVC. Since VEVC is a decisive factor to describe the vertical mixing in the ocean, it is necessary to pay attention to the inversion of the VEVC. The other objective is to make a computational investigation on the performance of the gradient descent method and the limited-memory Broyden-Fletcher-Goldfarb-Shanno (L-BFGS) method for the inversion of VEVCs based on the model constructed by Chen et al. [27]. Both of the methods do not require any evaluations of the Hessian matrices but gradient vectors and, thus, are computationally feasible. Chen et al. [30] have made a comparative study on several optimization methods but it is on the inversion of the open boundary conditions which is a one-dimension case. The feasibility of these optimization methods for two-dimensional case such as the inversion of the VEVC needs further studies.

Two groups of numerical experiments are carried out to study the inversion of spatially varying VEVCs based on an isopycnic-coordinate internal tidal model with the adjoint method. In Group One, the influences of independent point schemes (IPSs) exerting on parameter inversion are discussed. Group Two investigates the inversions of VEVCs on two different Gaussian bottom topographies and the performances of two optimization methods which are the GDM-S and the L-BFGS methods.

This paper is organized as follows. Section 2 briefly introduces the adjoint tidal model and the methodology. Two optimization methods, including the GDM-S and the L-BFGS methods, are described in Section 3. Section 4 presents design and process of the experiments in detail. Results of the experiments are discussed in Section 5. Finally, we make a summary and draw some conclusions in Section 6.

2. Numerical Model Introduction

An isopycnic-coordinate internal tidal model with adjoint assimilation method is employed in this paper. There are two parts in the internal tide model. One is forward model with the governing equations and the other is adjoint model with the adjoint equations. The two models are used to simulate the internal tide and to optimize the control variables, respectively. Chen et al. [27] had introduced the two parts in great detail and tested the reasonability and feasibility of the model. The formulation will not be presented in this paper. The derivation of VEVC adjustment, introduction of the two optimization method, test of the adjoint method, and the independent point scheme (IPS) are described in details in this part.

2.1. Test of the Adjoint Method. According to the equations and derivations of Chen et al. [27], the formula to invert the VEVC can be derived. The first derivative of Lagrangian function with respect to VEVC is obtained as follows:

$$\frac{\partial L}{\partial A_{vi,j,k}} = 0, \tag{1}$$

where $A_{vi,j,k}$ is the value of VEVC at grid (i, j) in the kth layer. The gradient of cost functions with respect to the VEVC in the grid (i, j, k) can be deduced as follows:

$$\begin{aligned}
&\frac{\partial J}{\partial A_{vi,j,k}} \\
&\quad = (\rho_k + \rho_{k+1}) \\
&\qquad \times \sum_n \left[\frac{\left(u^n_{i,j,k} - u^n_{i,j,k+1}\right)\left(u^n_{ai,j,k} - u^n_{ai,j,k+1}\right)}{h_{i,j,k} + h_{i+1,j,k} + h_{i,j,k+1} + h_{i+1,j,k+1}} \right. \\
&\qquad\qquad \left. + \frac{\left(u^n_{i-1,j,k} - u^n_{i-1,j,k+1}\right)\left(u^n_{ai-1,j,k} - u^n_{ai-1,j,k+1}\right)}{h_{i-1,j,k} + h_{i,j,k} + h_{i-1,j,k+1} + h_{i,j,k+1}} \right] \\
&\quad + (\rho_k + \rho_{k+1}) \\
&\qquad \times \sum_n \left[\frac{\left(v^n_{i,j,k} - v^n_{i,j,k+1}\right)\left(v^n_{ai,j,k} - v^n_{ai,j,k+1}\right)}{h_{i,j,k} + h_{i+1,j,k} + h_{i,j,k+1} + h_{i+1,j,k+1}} \right. \\
&\qquad\qquad \left. + \frac{\left(v^n_{i-1,j,k} - v^n_{i-1,j,k+1}\right)\left(v^n_{ai-1,j,k} - v^n_{ai-1,j,k+1}\right)}{h_{i-1,j,k} + h_{i,j,k} + h_{i-1,j,k+1} + h_{i,j,k+1}} \right],
\end{aligned} \tag{2}$$

where ρ_k is the potential density in the kth layer, $u^n_{i,j,k}$ and $v^n_{i,j,k}$ are horizontal velocities at the nth time step, $u^n_{ai,j,k}$ and $v^n_{ai,j,k}$ are the adjoint variables of $u^n_{i,j,k}$ and $v^n_{i,j,k}$, respectively, and $h_{i,j,k}$ is the initial thickness of the kth layer. The detailed derivation of (2) is presented in the appendix.

Accurately programming the adjoint in such problems as the present one is quite tricky and experience has shown that it is essential to check the accuracy of the adjoint computation before proceeding with the minimization runs [33]. The correctness of the adjoint method is verified in this section. Take the first-order term of a Taylor expansion for the cost function and we obtain the following equation:

$$J(\mathbf{p} + \alpha\mathbf{U}) = J(\mathbf{p}) + \alpha\mathbf{U} \cdot G(\mathbf{p}) + O\left(\alpha^2\right). \tag{3}$$

Here, $\mathbf{p}$ is a general point of the control variable, $G(\mathbf{p}) = \nabla J(\mathbf{p})$ is the computed gradient, and $\mathbf{U}$ is an arbitrary unit vector in the parameter space. Based on (3) a function of α can be written as follows:

$$\Phi(\alpha) = \frac{J(\mathbf{p} + \alpha\mathbf{U}) - J(\mathbf{p})}{\alpha\mathbf{U} \cdot G(\mathbf{p})}, \tag{4}$$

where α is a small real number that is not equal to zero. If the adjoint methodology is correct, it is supposed that $\lim_{\alpha \to 0}\Phi(\alpha) = 1$ according to (4). In this paper, the VEVC variable A_v is treated as $\mathbf{p}$ and test of the adjoint method is based on (4).

In order to test the accuracy of the adjoint method, two experiments are carried out in which two different types of $\mathbf{U}$ are used. The different vector directions are $U_1 = G(p)/|G(p)|$ and $U_2 = (\sqrt{1/N}, \sqrt{1/N}, \ldots, \sqrt{1/N})$, respectively.

Figure 1 indicates the trends of $\Phi(\alpha)$ as α approaches to 0. It is clear that in both experiments when α is less than 10^{-3}, values of Φ (solid lines) are both very close to 1 (dashed lines). Equation $\lim_{\alpha \to 0}\Phi(\alpha) = 1$ is proved and the correctness of gradient computed in the adjoint model is verified.

2.2. Independent Point Scheme. The available observation data may not be sufficient and control parameters to be determined may be excessive in practice. That may cause ill-posedness of the inversion problem. Richardson and Panchang [34] first noted that if an adjoint method is applied, when there is a big error in data, the solution will be unstable and not unique. Many researchers have made progress in solving this problem. A lot of work [26, 27, 30, 32] have been done to prove the capability and feasibility of the independent point scheme (IPS) in solving ill-posed problems of inversion.

In this paper, the IPS is used to optimize the control parameter. The basic idea of IPS is as follows: some grids (e.g., (ii, jj)) are selected as the independent points; it is assumed that $P_{ii,jj}$ represents the value of VEVC in grid (ii, jj), so values of VEVC in all grids $p_{i,j}$ can be calculated from $P_{ii,jj}$ with linear interpolation method. The computing formula is given as follows:

$$p_{i,j} = \frac{\sum_{ii,jj} W_{i,j,ii,jj} P_{ii,jj}}{\sum_{ii,jj} W_{i,j,ii,jj}}, \tag{5}$$

where $W_{i,j,ii,jj}$ is the weight coefficient of the Cressman form [35]:

$$W_{i,j,ii,jj} = \frac{R^2 - r^2_{i,j,ii,jj}}{R^2 + r^2_{i,j,ii,jj}}, \tag{6}$$

where $r_{i,j,ii,jj}$ is the center distance between (i, j) and (ii, jj) and R is the influence radius. According to (5), the gradient of J with respect to $p_{i,j}$ can be written as

$$\frac{\partial J}{\partial P_{ii,jj}} = \sum_{i,j} W_{i,j,ii,jj} \frac{\partial J}{\partial p_{i,j}} \sum_{i,j} W_{i,j,ii,jj}, \tag{7}$$

where $\partial J/\partial p_{i,j}$ is the gradient of J with respect to the VEVC at the grid (i, j) and can be calculated using formula (2).

According to Section 2.1, the correctness of the gradient with respect to the independent points should be tested. Based on (5), the perturbations (αU) applied to the independent points have a linear impact on those applied to the nonindependent points. The convergences for the nonindependent points will remain the same after linear transformation for independent points.

The values of VEVC at the independent grids can be calculated inside the model and values at other grids are

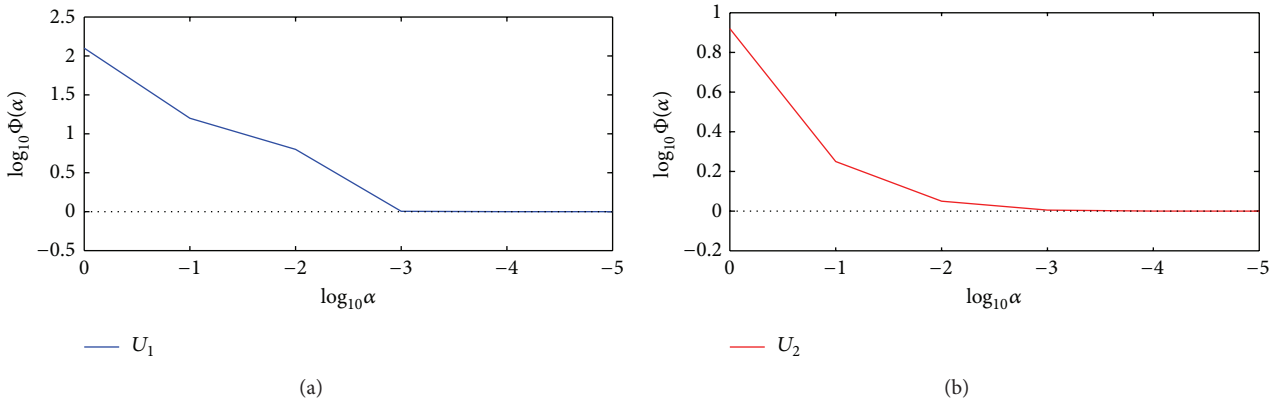


Figure 1: Variation of $\Phi(\alpha)$ with respect to α.

gained through interpolation using (5). Then the spatial distribution of VEVC in the entire area is obtained.

3. Optimization Algorithms

There have been many large-scale optimization methods to solve the minimization problem [36]. Four main methods are line search method (e.g., Wolfe and Goldstein), trust-region method, conjugate gradient method (e.g., Fletcher-Reeves), and quasi-newton method (e.g., BFGS and L-BFGS). However, the number of studies discussing the performances of various optimization methods in the meteorological and oceanographic application is still relatively small [30]. Line search method requires repeated computations for the cost function and the gradient. Therefore it spends too much computation resource during numerical simulations especially for physical oceanography. Besides, line search methods may fail in some cases [37]. The L-BFGS method is commonly used to solve large-scale problems in oceanography and meteorology [30, 38, 39]. Chen et al. [30] have made a computational investigation on the performances of the L-BFGS method and two versions of gradient descent method with an internal tide model. In their work, a simplified gradient descent method is applied which is able to avoid too much computation. The step length is chosen according to the experience of the modeler. Compared with other methods, there are two main advantages of this plan. One is the less computation resource usage and the other is the more controllable optimization process. Many research papers have proved the feasibility of this method [25–27, 31, 32].

Generally speaking, numerical methods to solve the minimization problems have the similar iterative formula as follows:

$$A_v^{n+1} = A_v^n + \alpha^n d^n, \tag{8}$$

where A_v^n and A_v^{n+1} are the priori and adjusted values of VEVC in the nth iteration, respectively and α^n and d^n represent the iteration step length and the search direction, respectively. There are many methods to determine the search direction d_n. Two different optimization methods employed in this paper are the GDM-S and the L-BFGS methods.

3.1. Gradient Descent Method (GDM). The GDM is a simple and feasible method to define the search direction as follows:

$$d_n = -g_n = -\frac{\left(\partial J/A_{vi,j,k}\right)_n}{\left\|\left(\partial J/A_{vi,j,k}\right)_n\right\|}, \tag{9}$$

where i, j, and k are the zonal index, the meridional index, and the layer index of the calculation grid, respectively; n represents the step of iteration. $\|(\partial J/A_{vi,j,k})_n\|$ is the L_2 norm of the gradient of the cost function with respect to the VEVC in the nth iteration.

In the GDM-S, the step length α is chosen to be a constant according to the experience of the modeler. We have surveyed the performances of different values of α and take 0.006 as the best choice. The optimized value of VEVC is obtained after the VEVC in every grid (i, j, k) is adjusted according to (8) and (9).

3.2. L-BFGS Method. L-BFGS is an optimization algorithm in the family of quasi-Newton methods that approximates the BFGS algorithm using a limited amount of computer memory. This method is first described in the work of Nocedal [40], where it is called the SQN method. It is a popular algorithm for parameter estimation in machine learning [41, 42]. Due to its resulting linear memory requirement, the L-BFGS method is particularly well suited for optimization problems with a large number of variables.

It requires the search direction to be

$$d_k = -H_k g_k, \tag{10}$$

where

$$\begin{aligned} H_{k+1} = {} & \left(V_{k-1}^T \cdots V_{k-m}^T\right) H_0 \left(V_{k-m} \cdots V_{k-1}\right) \\ & + \rho_{k-m} \left(V_k^T \cdots V_{k-m+1}^T\right) s_{k-m} s_{k-m}^T \left(V_{k-m+1} \cdots V_k\right) \\ & + \rho_{k-m} \left(V_k^T \cdots V_{k-m+1}^T\right) s_{k-m} s_{k-m}^T \left(V_{k-m+1} \cdots V_k\right) \\ & + \rho_{k-m} \left(V_k^T \cdots V_{k-m+1}^T\right) s_{k-m} s_{k-m}^T \left(V_{k-m+1} \cdots V_k\right) \\ & \vdots \\ & + \rho_k s_k s_k^T . \end{aligned} \tag{11}$$

Note that g_k is the simplification of $\nabla J(p_k)$ for convenience. Here, $\rho_k = 1/s_k y_k^T$, $V_k = 1 - \rho_k y_k s_k^T$, $s_k = p_{k+1} - p_k = \alpha_k d_k$, and $y_k = g_{k+1} - g_k$. Many studies have shown that typically $3 \le m \le 7$, where $m > 7$ does not improve the performance of L-BFGS [43]. So the number of corrections m used in the L-BFGS update of this paper is taken as 5 [44]. The version of L-BFGS used in this paper is described in Liu and Nocedal [44] and the Fortran codes are authorized by Nocedal [45].

4. Design of Experiments

All the experiments in this paper are implemented in an ideal regional area from 116°E to 124.17°E and from 18°N to 23.17°N with in mind the practical sea area located around the Luzon strait. The horizontal resolution is $10' \times 10'$ and there are totally 49×31 grids in the area. The maximum depth is set to be 1000 meters. The horizontal eddy coefficient is chosen to be $A_h = 1000$ and the bottom friction coefficient is taken as $\kappa = 0.003$. The Coriolis coefficient is taken as the local value. Only M_2 tide is considered and its angular frequency is $1.41 \times 10^{-4}\ \mathrm{s}^{-1}$. The whole-time step is 496.86 s which is 1/90 of the period of M_2 tide. All the four boundaries are open boundaries and boundary conditions are set to be the local water levels in the Flather form.

Eastern and western boundaries:

$$\zeta - \widetilde{\zeta}' = \pm \sqrt{\left(1 - \frac{f^2}{\omega^2}\right) \frac{H}{g}} \left(U - \widetilde{U}\right), \tag{12}$$

positive in eastern boundary and negative in western boundary.

North and south boundaries:

$$\zeta - \widetilde{\zeta}' = \pm \sqrt{\left(1 - \frac{f^2}{\omega^2}\right) \frac{H}{g}} \left(V - \widetilde{V}\right), \tag{13}$$

positive in north boundary and negative in south boundary.

ζ is the surface elevation above the undisturbed sea level. U and V are the zonal current velocity and the meridional current velocity, respectively. $\widetilde{\zeta}'$, $\widetilde{U}$, and $\widetilde{V}$ are the surface elevation and the zonal and meridional current velocity relating to the boundary barotropic tidal force, respectively. f is the Coriolis coefficient and ω is the tidal frequency of M_2 tide. H is the local water depth and g is the acceleration of gravity.

Similar as Chen et al. [27], the M_2 tidal force at the nth time step is subject to

$$\widetilde{\zeta}'^2 = a_\zeta \cos(\omega n \Delta t) + b_\zeta \sin(\omega n \Delta t), \tag{14}$$

the Fourier coefficients a_ζ at four open boundaries are set to be 0 and b_ζ are set to be 1 (−1) at the north and west (south and east) boundaries.

The T/P altimeter data is widely spread throughout the ocean and it can be used to invert VEVC. In this work, we pick 89 calculating points based on the distribution features of T/P altimeter observation as the observation points (Figure 2).

Two kinds of topographies are tested in this paper and they are generated based on the two formulas in (15), respectively (Figure 3). The sea water in the computing area is divided into two layers. The thicknesses of each layers are, respectively, $h_1 = 200$ m and $h_2 = 800$ m. The potential densities of corresponding layer are $\rho_1 = 1021\ (\mathrm{kg/m^{-3}})$ and $\rho_2 = 1024\ (\mathrm{kg/m^{-3}})$, respectively. Consider

$$\begin{aligned} H_A &= h_0 \exp\left(-\frac{(x - x_0)^2 + (y - y_0)^2}{2L^2}\right) \\ &\quad - \mathrm{MaxDepth}, \\ H_B &= h_0 \left(1 - \exp\left(-\frac{(x - x_0)^2 + (y - y_0)^2}{2L^2}\right)\right) \\ &\quad - \mathrm{MaxDepth}, \end{aligned} \tag{15}$$

where h_0 is the height of the topography and MaxDepth is the depth of the water.

For each experiment, the optimization of the VEVC can be implemented with the following steps.

Step 1. The prescribed VEVC is given and the forward model is run. The whole simulation time is 20 period of the M_2 tide in order to obtain a stable simulation result. The water elevations in the observation positions are treated as the "pseudoobservations."

Step 2. Initial value of the control parameter (VEVC) is given and forward model is run to get the simulated results of all the state variables such as current velocity and water elevation. The value of cost function J is calculated.

Step 3. Difference between the simulated elevation and the "pseudoobservation" plays as the external force of the adjoint model. Via backward integrating the adjoint equations in a period of the M_2 tide values of the adjoint variables are obtained.

Step 4. Using formula (2), along with the state variables and the adjoint variables obtained in Steps 2 and 3, the gradient of cost function with respect to VEVC is calculated.

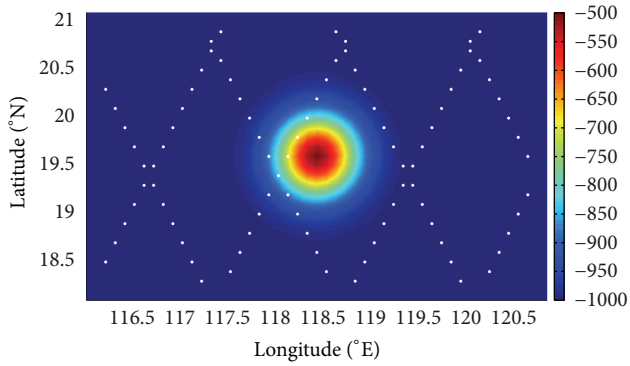


Figure 2: Planform of topography (e.g., topography A) and locations of the observations (white dots).

Step 5. Update the unknown control variables with a certain optimization method.

Step 6. If the stopping criterion of iteration is reached, bring the iteration to an end and return the optimized parameter. Otherwise, update all the parameters and go back to Step 2.

In the experiments of this paper, all initial values of VEVC are set to 0.005 and the total number of iterations is allowed to be 100 at most. The chosen convergence criterion is that the last two values of the cost function are sufficiently close, which is defined by

$$\left|J_{\text{end}} - J_{\text{end}-1}\right| < 10^{-9}, \tag{16}$$

where J_{end} and $J_{\text{end}-1}$ are the last and the second last values of the cost function, respectively.

Two groups of numerical experiments are carried out: the influence of IPSs on the inversion of VEVC is studied in Group One; in Group Two the ability of this internal tide model to invert different kinds of VEVC with spatial distribution is examined. Two kinds of spatial distribution of VEVC are prescribed and given in Figure 4. For both distributions, the VEVC value ranges from 3×10^{-3} m^2/s to 7×10^{-3} m^2/s.

In Group One, nine experiments are carried out to discuss the influence of IPS on the inversion of VEVC. The distance between independent points (IP) ranges from $20'$ (length of 2 grids) to $100'$ (length of 10 grids) and details are listed in Table 1. Topography A is applied and distribution (a) is chosen to be the prescribed spatial distribution of VEVC.

In Group Two, four numerical experiments (NEs) are carried out which are numbered as NE1~NE4, respectively. Each experiment is implemented with GDM-S and the L-BFGS methods. Information of topographies and prescribed VEVCs for all NEs is listed in Table 2. The IPSs are the optimal schemes obtained in Group One. Other parameters are exactly the same as Group One.

All the experiments in Group One and Group Two are carried out following Steps 1 to 6 described in Section 4. Version of L-BFGS method used in this paper is from the work of Liu and Nocedal [44].

Table 1: Settings of independent point schemes in Group One.

IPS	1	2	3	4	5	6	7	8	9
Number of IPs	360	160	96	60	40	35	24	24	15
Distance between IPs ($'$)	20	30	40	50	60	70	80	90	100

Table 2: Topographies and distributions of VEVC in Group Two.

Experiment	Topography	Distribution
NE1	A	a
NE2	A	b
NE3	B	a
NE4	B	b

5. Experiment Results and Discussions

5.1. Results and Discussions of Group One. Figure 5 illustrates the relationships between mean absolute errors (MAE, which reflects the error between the inversion result and the given VEVC) and the distance between independent points in Group One.

As is shown in Figure 5, MAEs with different optimization methods vary as the IPSs change. The minimum MAEs with the two methods are not the same (dashed lines). The minimum MAE with the GDM-S is 2.66×10^{-4} m^2/s while that with the L-BFGS method reaches 9.14×10^{-5} m^2/s. The corresponding IP distances are $90'$ (length of 9 grids, GDM-S) and $30'$ (length of 3 grids, L-BFGS), respectively. According to the MAEs illustrated in Figure 5, the optimal IPSs of the two methods are selected as IPS8 (GDM-S) and IPS 2 (L-BFGS).

5.2. Results and Discussions of Group Two. With the respective optimal IPSs and the iteration process in Section 4, the VEVCs are inverted with GDM-S method (I) and L-BFGS method (II), respectively, and the inversion results are shown in Figure 6.

Comparison of the inversion results with the prescribed VEVCs indicates that all the given spatial distributions of VEVC are successfully inverted after 100 iteration steps. The main features of all distributions can be recovered very well. Surfaces of the inverted VEVC with the L-BFGS are much smoother than those with the GDM-S. Compared against the inversion results with the GDM-S (left panels), patterns with the L-BFGS method (right panels) are closer to the prescribed VEVC. More statistic data will be presented in the next paragraphs.

MAEs of the four numerical experiments after assimilation are calculated and listed in Table 3. The initial values of MAE in all NEs are 1×10^{-3} m^2/s. Based on the results shown in Table 3, all the MAEs after assimilation are more than one order of magnitude lower than the initial values, which means the success for both of the two optimization methods. Note that the MAEs with the L-BFGS method are quite close (less than 1×10^{-8} m^2/s and cannot be distinguished in the table) and are less than half of those with the GDM-S. This result demonstrates the ability of the L-BFGS method to deduce the overall errors.

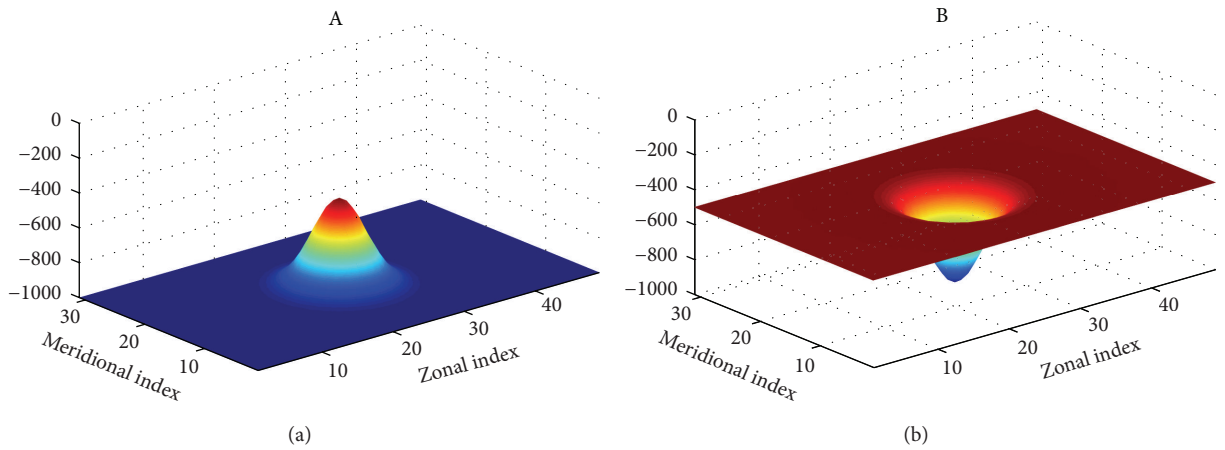


Figure 3: Topographies A and B. Note that the abscissa and ordinate axes are labeled with zonal index and meridional index, respectively.

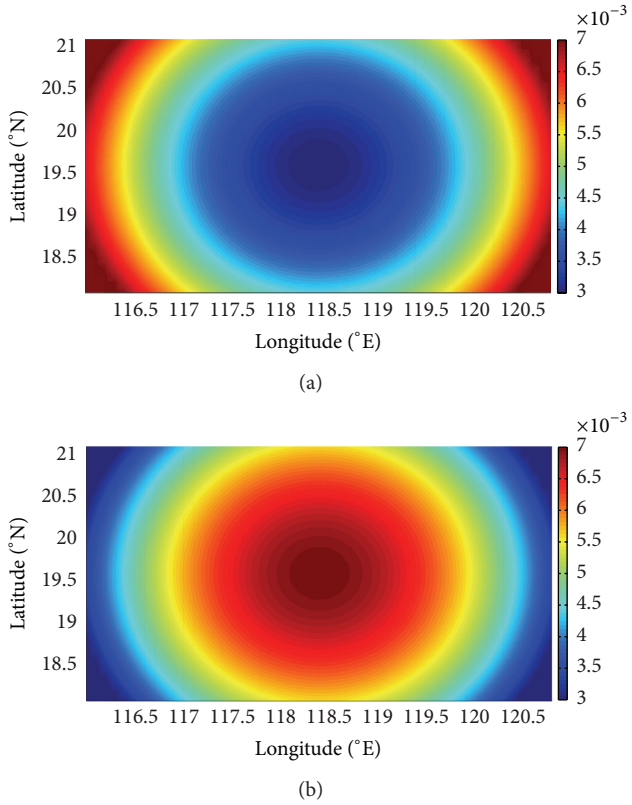


Figure 4: Planform of two prescribed spatial distributions of VEVC.

Table 3: Inversion errors of VEVC in Group Two (unit: m^2/s).

Method	Experiment			
	NE1	NE2	NE3	NE4
GDM-S(I)	$2.42E-04$	$2.43E-04$	$2.23E-04$	$2.24E-04$
L-BFGS(II)	$9.14E-05$	$9.14E-05$	$9.14E-05$	$9.14E-05$

To compare the effectiveness of the two methods to invert the VEVC, we make statistics on the percentages of the grids

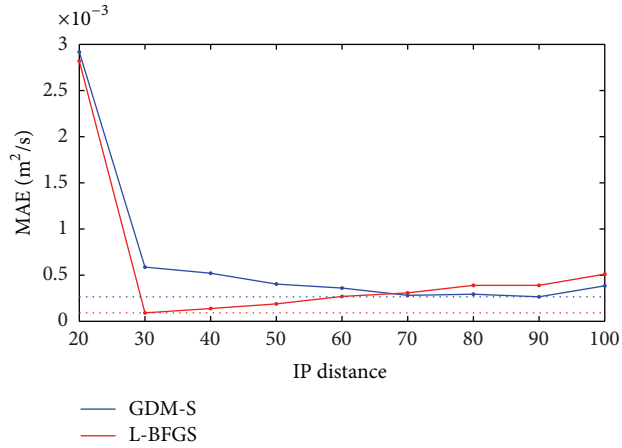
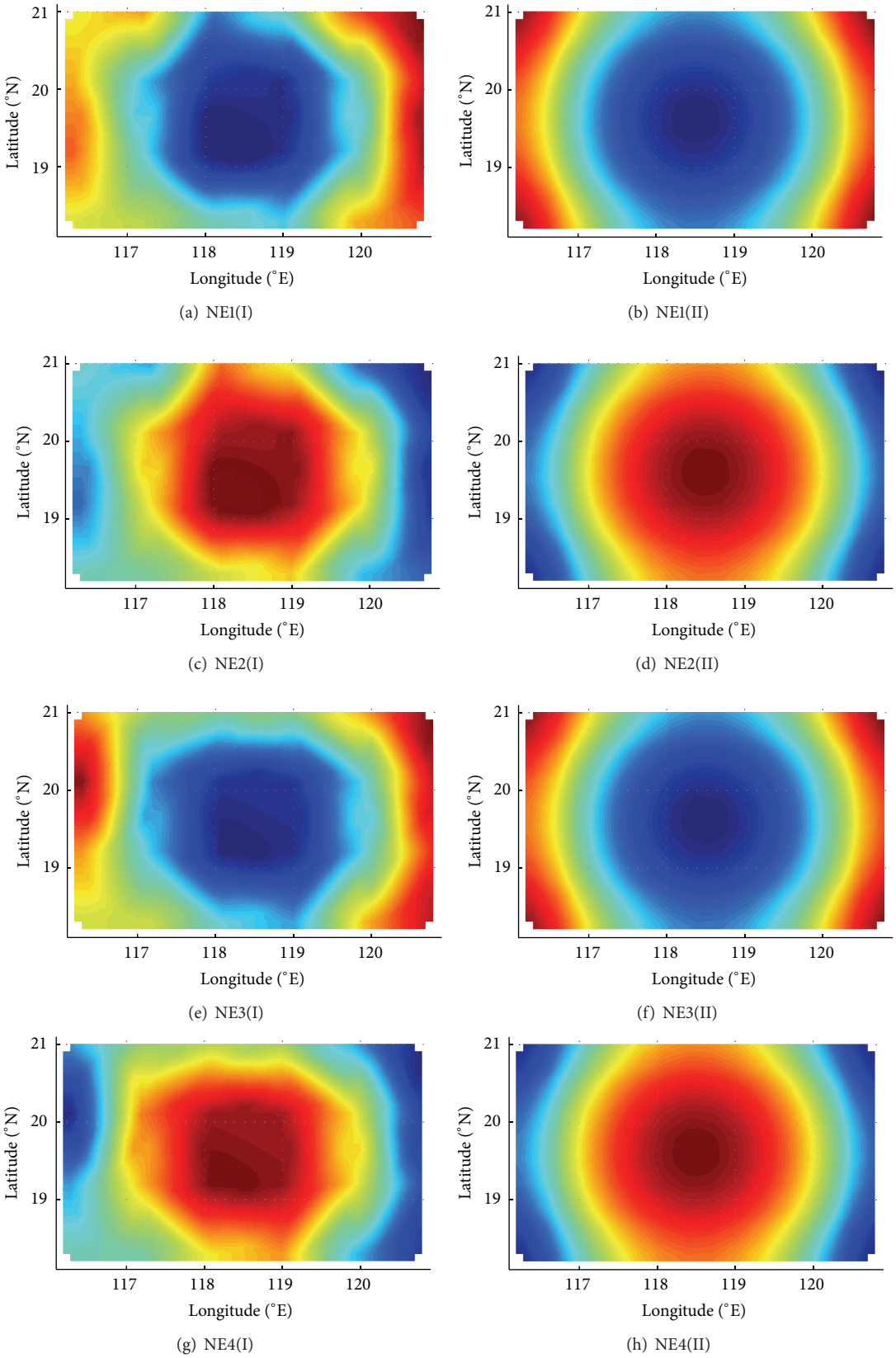


Figure 5: MAEs versus IP distance in Group One. The abscissas indicates distance between adjacent IPs (unit: $'$) while the ordinate indicates MAE of inversion results. The solid lines are values of different experiments and the dashed lines indicate the minimum values of two solid lines, respectively.

at which the MAEs are less than $1 \times 10^{-4}\ m^2/s$, which is listed in Table 4. With the GDM-S, the inversion errors are deduced by one order of magnitude at about 40% of the total grids. By contrast, ratios of all NEs with the L-BFGS method are 79.76%, without differences between NEs. This phenomenon indicates that the L-BFGS method is effective at more computation grids than the GDM-S. Furthermore, the L-BFGS method maintains its effectiveness no matter which topography is applied.

Combining the inversion patterns, the inversion errors of the VEVC, and the effectiveness analyses, conclusions can be drawn that the L-BFGS has a better performance in reducing the inversion errors.

Finally we come to the optimization history for all the experiments carried out in Group Two. The variations of the cost function normalized by its initial value, that of the L_2 norm of the gradient of the cost function with respect to the



(a) NE1(I) (b) NE1(II)

(c) NE2(I) (d) NE2(II)

(e) NE3(I) (f) NE3(II)

(g) NE4(I) (h) NE4(II)

Figure 6: Planform of inversion results in Group Two.

Table 4: Effectiveness analyses of inversions in Group Two.

Method	Experiment			
	NE1	NE2	NE3	NE4
GDM-S(I)	42.53%	42.24%	40.99%	40.91%
L-BFGS(II)	79.76%	79.76%	79.76%	79.76%

VEVCs and that of the inversion error, are plotted in Figures 7(a), 7(b), and 7(c), respectively, as a function of the iteration step.

Note that all experiments with the L-BFGS method reach the convergence criterion and stop after 4 iterations, which indicates that the computation time for the L-BFGS method is one twenty-fifth of that for the GDM-S. Figure 7(a) indicates that all the cost functions are in downward trends throughout the iteration process and decrease by more than 2 (7) orders of magnitude for the GDM-S (L-BFGS method), which means that the final differences between simulation value and the observation of these two methods are less than one-tenth and one-thousandth of their initial values, respectively. As is shown in Figure 7(b), the L_2 norms of gradient of the cost function with respect to the VEVC decrease by more than 1 order of magnitude (GDM-S) and 4 orders of magnitude (L-BFGS), compared with their respective initial values. This indicates that the inversion result is becoming increasingly closer to the given VEVC during the iteration. As shown in Figure 7(c), the inversion errors with the two methods keep declining throughout the iterations until the stopping criterions are satisfied. In general, the cost functions, norms of gradient, and the inversion errors of the VEVC have steady descent, which demonstrates that this adjoint model is capable to invert the VEVC. What is more, both the GDM-S and the L-BFGS methods are effective in terms of the inversion of the control parameters with spatially distributions of internal tide.

It is also clear in Figure 7 that the convergence rate for the cost functions, norms of gradient, and the inversion errors of the VEVC is much faster with the L-BFGS method than those with the GDM-S, which is consistent with the classic theories about the convergence rate of the quasi-Newton method and the GDM [36]. This trend is also consistent with the results of numerical experiments to invert the open boundary conditions in Chen et al. [30]. With no doubt, the L-BFGS method is a more effective and efficient optimization method to invert the spatially varying VEVC. However, the GDM-S is easier to understand and to implement in the model. Moreover, the step length and the search direction in the process of GDM-S can be freely controlled by the modelers, which is very convenient in practice. Therefore, for the inversion of the VEVC, the GDM-S should also be regarded as a choice.

6. Summary and Conclusions

Based on an isopycnic-coordinate internal tidal model, the inversion of VEVC is studied in this paper. A series of numerical experiments are carried out to examine the influence factors on the inversion of VEVCs in four aspects: independent point schemes (IPS), topography, the spatial distribution of VEVC, and the optimization methods. For each experiment, the cost function, the L_2 norm of gradient of cost function with respect to the VEVC, and the inversion error are calculated and analyzed in details.

The IPS is introduced and discussed in Group One. All the VEVCs can be inverted successfully with IPS. MAE is regarded as the comparison criterion of the result. After comparing the 9 experiments, the correctness of the IPS is confirmed and the optimal IPSs are selected for the GDM-S and the L-BFGS methods, respectively.

Based on the optimal IPSs in Group One, two kinds of VEVC distributions are successfully inverted with this adjoint model on two kinds of topography in Group Two. MAEs after optimization are at the level of 10^{-4} (10^{-5}) for the GDM-S (L-BFGS), which is one (two) order(s) of magnitude lower than the initial value. All the cost functions and their gradient norms with respect to the VEV lead satisfactory declines no matter which optimization method is taken. Compared with the GDM-S, the L-BFGS method has a remarkably better performance, not only in terms of the convergence rate but also in terms of the final inversion results. The computation time for the L-BFGS method is much shorter than that for the GDM-S. To sum up, the L-BFGS method is a more effective and efficient method than the GDM-S in terms of the inversion of the VEVC. Nevertheless, the GDM-S is more convenient and controllable so it should not be ignored and should be taken seriously as a choice for the inversion of the VEVC with spatially distribution.

The success of numerical experiments lays a solid foundation for the practical experiments and encourages us to carry out experiments in practical sea area with measured data and the real T/P altimeter data.

Appendix

Derivation of (2)

Let us start with the governing equations in Chen et al. [27].

Layer 1 (surface layer)

$$\frac{\partial q_1}{\partial t} + \frac{1}{R\cos\varphi}\frac{\partial (q_1 u_1)}{\partial \lambda} + \frac{1}{R\cos\varphi}\frac{\partial (q_1 v_1 \cos\varphi)}{\partial \varphi} = 0, \tag{A.1a}$$

$$\frac{\partial u_1}{\partial t} + \frac{u_1}{R\cos\varphi}\frac{\partial u_1}{\partial \lambda} + \frac{v_1}{R}\frac{\partial u_1}{\partial \varphi} - \frac{u_1 v_1 \tan\varphi}{R} - f v_1 - A_{h1}\Delta u_1 + \frac{\tau_{1\lambda}}{q_1} + \frac{g}{R\cos\varphi}\left[\frac{\partial}{\partial \lambda}\sum_{m=1}^{l}\left(\frac{q_m}{\rho_m} - h_m\right) + \frac{\rho_1}{\rho_k}\frac{\partial \bar{\zeta}}{\partial \lambda}\right] = 0, \tag{A.1b}$$

$$\frac{\partial v_1}{\partial t} + \frac{u_1}{R\cos\varphi}\frac{\partial v_1}{\partial \lambda} + \frac{v_1}{R}\frac{\partial v_1}{\partial \varphi} + \frac{u_1^2 \tan\varphi}{R} - f u_1 - A_{h1}\Delta v_1 + \frac{\tau_{1\varphi}}{q_1} + \frac{g}{R}\left[\frac{\partial}{\partial \varphi}\sum_{m=1}^{l}\left(\frac{q_m}{\rho_m} - h_m\right) + \frac{\rho_1}{\rho_k}\frac{\partial \bar{\zeta}}{\partial \varphi}\right] = 0. \tag{A.1c}$$

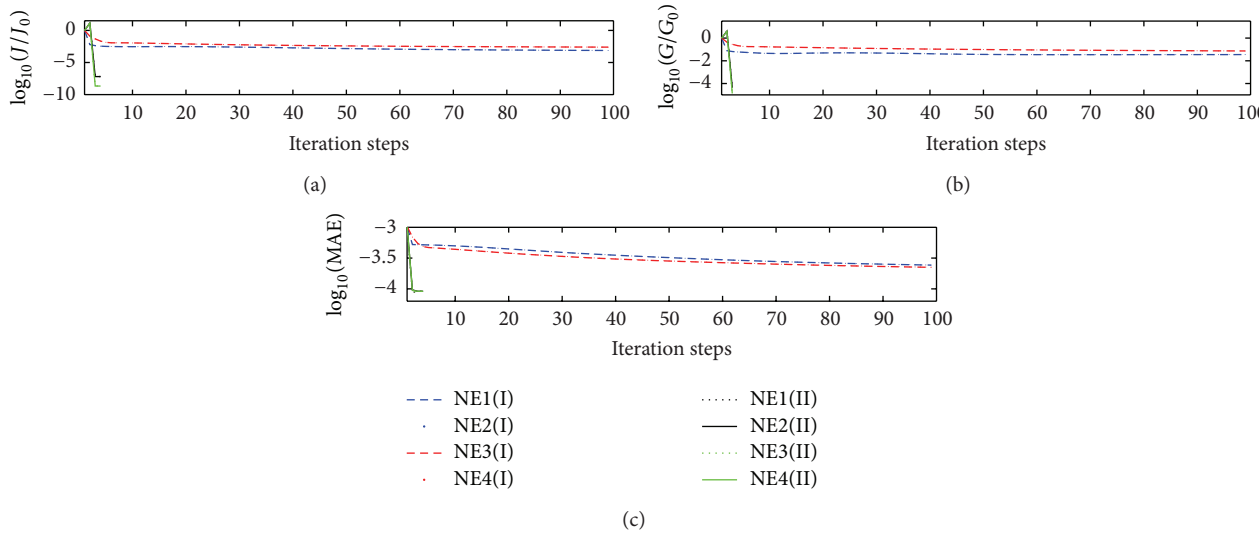


FIGURE 7: Optimization history for experiments of Group Two, about (a) the cost function normalized by its initial value J_0, (b) the L_2 norm of gradient of the cost function with respect to the VEVC, and (c) the MAEs between the inverted and prescribed VEVCs.

Layer k $(k = 2, \ldots, l-1)$

$$\frac{\partial q_k}{\partial t} + \frac{1}{R\cos\varphi}\frac{\partial (q_k u_k)}{\partial \lambda} + \frac{1}{R\cos\varphi}\frac{\partial (q_k v_k \cos\varphi)}{\partial \varphi} = 0, \tag{A.2a}$$

$$\begin{aligned}&\frac{\partial u_k}{\partial t} + \frac{u_k}{R\cos\varphi}\frac{\partial u_k}{\partial \lambda} + \frac{v_k}{R}\frac{\partial u_k}{\partial \varphi} - \frac{u_k v_k \tan\varphi}{R} - f v_k - A_{hk}\Delta u_k \\ &- \frac{\tau_{(k-1)\lambda} - \tau_{k\lambda}}{q_k} + \frac{g}{R\cos\varphi} \\ &\times \left[\sum_{m=1}^{k-1}\left(\frac{1}{\rho_k} - \frac{1}{\rho_m}\right)\frac{\partial q_m}{\partial \lambda} \right. \\ &\left. + \frac{\partial}{\partial \lambda}\sum_{m=1}^{l}\left(\frac{q_m}{\rho_m} - h_m\right) + \frac{\rho_1}{\rho_k}\frac{\partial \bar{\zeta}}{\partial \lambda}\right] = 0,\end{aligned} \tag{A.2b}$$

$$\begin{aligned}&\frac{\partial v_k}{\partial t} + \frac{u_k}{R\cos\varphi}\frac{\partial v_k}{\partial \lambda} + \frac{v_k}{R}\frac{\partial v_k}{\partial \varphi} + \frac{u_k^2 \tan\varphi}{R} + f u_k \\ &- A_{hk}\Delta v_k - \frac{\tau_{(k-1)\varphi} - \tau_{k\varphi}}{q_k} + \frac{g}{R} \\ &\times \left[\sum_{m=1}^{k-1}\left(\frac{1}{\rho_k} - \frac{1}{\rho_m}\right)\frac{\partial q_m}{\partial \varphi} \right. \\ &\left. + \frac{\partial}{\partial \varphi}\sum_{m=1}^{l}\left(\frac{q_m}{\rho_m} - h_m\right) + \frac{\rho_1}{\rho_k}\frac{\partial \bar{\zeta}}{\partial \varphi}\right] = 0.\end{aligned} \tag{A.2c}$$

Layer l (bottom layer)

$$\frac{\partial q_l}{\partial t} + \frac{1}{R\cos\varphi}\frac{\partial (q_l u_l)}{\partial \lambda} + \frac{1}{R\cos\varphi}\frac{\partial (q_l v_l \cos\varphi)}{\partial \varphi} = 0, \tag{A.3a}$$

$$\begin{aligned}&\frac{\partial u_l}{\partial t} + \frac{u_l}{R\cos\varphi}\frac{\partial u_l}{\partial \lambda} + \frac{v_l}{R}\frac{\partial u_l}{\partial \varphi} - \frac{u_l v_l \tan\varphi}{R} - f v_l \\ &- A_{hl}\Delta u_l - \frac{\tau_{(l-1)\lambda} - \tau_{b\lambda}}{q_l} + \frac{g}{R\cos\varphi} \\ &\times \left[\frac{\partial}{\partial \lambda}\sum_{m=1}^{l}\left(\frac{q_m}{\rho_l} - h_m\right) + \frac{\rho_l}{\rho_l}\frac{\partial \bar{\zeta}}{\partial \lambda}\right] = 0,\end{aligned} \tag{A.3b}$$

$$\begin{aligned}&\frac{\partial v_l}{\partial t} + \frac{u_l}{R\cos\varphi}\frac{\partial v_l}{\partial \lambda} + \frac{v_l}{R}\frac{\partial v_l}{\partial \varphi} + \frac{u_l^2 \tan\varphi}{R} + f u_l \\ &- A_{hl}\Delta v_l - \frac{\tau_{(l-1)\varphi} - \tau_{b\varphi}}{q_l} + \frac{g}{R} \\ &\times \left[\frac{\partial}{\partial \varphi}\sum_{m=1}^{l}\left(\frac{q_m}{\rho_l} - h_m\right) + \frac{\rho_l}{\rho_l}\frac{\partial \bar{\zeta}}{\partial \varphi}\right] = 0.\end{aligned} \tag{A.3c}$$

The variables and background of the governing equations have been introduced in Chen's [27] work in details. We will not repeat them in this part. The interface and friction terms are expressed by

$$\left(\tau_{k\lambda}, \tau_{k\varphi}\right) = A_{vk}\frac{\rho_{k+1/2}}{h_{k+1/2}}\left(u_k - u_{k+1}, v_k - v_{k+1}\right), \quad k = 1, \ldots, l-1, \tag{A.4}$$

where A_v is the vertical eddy viscosity coefficient, $\rho_{k+1/2} = (\rho_k + \rho_{k+1})/2$, and $h_{k+1/2} = (h_k + h_{k+1})/2$.

The cost function is defined as

$$J(q,u,v;\mathbf{p}) = \frac{1}{2}\int\left\{K_\varsigma\sum_{k=1}^{l}\left[\sum_{m=k}^{l}\left(\frac{q_m}{\rho_m}-h_m\right)\widehat{\zeta}_k\right]^2 + K_u\sum_{k=1}^{l}(u_k-\widehat{u}_k)^2 + K_v\sum_{k=1}^{l}(v_k-\widehat{v}_k)^2\right\}d\sigma, \tag{A.5}$$

which is exactly the same as that in [27]. Then the Lagrangian function is defined as

$$\begin{aligned} L(q,u,v;q_a,u_a,v_a;\mathbf{p}) &= J(q,u,v;\mathbf{p}) \\ &+ \int\left[q_{a1}\cdot(\text{A.1a}) + u_{a1}q_1\cdot(\text{A.1b}) + v_{a1}q_1\cdot(\text{A.1c})\right]d\sigma \\ &+ \cdots \\ &+ \int\left[q_{ak}\cdot(\text{A.2a}) + u_{ak}q_k\cdot(\text{A.2b}) + v_{ak}q_k\cdot(\text{A.2c})\right]d\sigma \\ &+ \cdots \\ &+ \int\left[q_{al}\cdot(\text{A.3a}) + u_{al}q_l\cdot(\text{A.3b}) + v_{al}q_l\cdot(\text{A.3c})\right]d\sigma, \end{aligned} \tag{A.6}$$

where

$$(\text{A.1a}) = -\left[\frac{\partial q_1}{\partial t} + \frac{1}{R\cos\varphi}\frac{\partial(q_1u_1)}{\partial\lambda} + \frac{1}{R\cos\varphi}\frac{\partial(q_1v_1\cos\varphi)}{\partial\varphi}\right], \tag{A.7}$$

$$(\text{A.1b}) = -\left\{\frac{\partial u_1}{\partial t} + \frac{u_1}{R\cos\varphi}\frac{\partial u_1}{\partial\lambda} + \frac{v_1}{R}\frac{\partial u_1}{\partial\varphi} - \frac{u_1v_1\tan\varphi}{R} - fv_1 - A_{h1}\Delta u_1 + \frac{\tau_{1\lambda}}{q_1} + \frac{g}{R\cos\varphi}\times\left[\frac{\partial}{\partial\lambda}\sum_{m=1}^{l}\left(\frac{q_m}{\rho_m}-h_m\right) + \frac{\rho_1}{\rho_k}\frac{\partial\bar{\zeta}}{\partial\lambda}\right]\right\}, \tag{A.8}$$

$$(\text{A.1c}) = -\left\{\frac{\partial v_1}{\partial t} + \frac{u_1}{R\cos\varphi}\frac{\partial v_1}{\partial\lambda} + \frac{v_1}{R}\frac{\partial v_1}{\partial\varphi} + \frac{u_1^2\tan\varphi}{R} - fu_1 - A_{h1}\Delta v_1 + \frac{\tau_{1\varphi}}{q_1} + \frac{g}{R}\times\left[\frac{\partial}{\partial\varphi}\sum_{m=1}^{l}\left(\frac{q_m}{\rho_m}-h_m\right) + \frac{\rho_1}{\rho_k}\frac{\partial\bar{\zeta}}{\partial\varphi}\right]\right\}, \tag{A.9}$$

same definitions are applied in (A.2a)~(A.2c) and (A.3a)~(A.3c). Then the Lagrangian function $L(q,u,v;q_a,u_a,v_a;\mathbf{p})$ can be written as

$$\begin{aligned} L(q,u,v;q_a,u_a,v_a;\mathbf{p}) &= J(q,u,v;\mathbf{p}) \\ &- \int\left[\frac{u_{a1}A_{vi+1/2,j,1}\rho_{1+1/2}}{h_{i+1/2,j,1+1/2}}(u_1-u_2) + \frac{u_{a1}A_{vi+1/2,j,1}\rho_{1+1/2}}{h_{i+1/2,j,1+1/2}}(v_1-v_2) - F_1\right]d\sigma \\ &+ \cdots \\ &- \int\left[\frac{u_{ak}A_{vi+1/2,j,k}\rho_{k+1/2}}{h_{i+1/2,j,k+1/2}}(u_k-u_{k+1}) - \frac{u_{ak}A_{vi+1/2,j,k-1}\rho_{k-1/2}}{h_{i+1/2,j,k-1/2}}(u_{k-1}-u_k)\right]d\sigma \\ &- \int\left[\frac{v_{ak}A_{vi+1/2,j,k}\rho_{k+1/2}}{h_{i+1/2,j,k+1/2}}(v_k-uv_{k+1}) - \frac{v_{ak}A_{vi+1/2,j,k-1}\rho_{k-1/2}}{h_{i+1/2,j,k-1/2}}(v_{k-1}-v_k) - F_k\right]d\sigma \\ &+ \cdots \\ &+ \int\left[\frac{u_{al}A_{vi+1/2,j,l-1}\rho_{l-1/2}}{h_{i+1/2,j,l-1/2}}(u_{l-1}-u_l) + \frac{u_{al}A_{vi+1/2,j,l-1}\rho_{l-1/2}}{h_{i+1/2,j,l-1/2}}(v_{l-1}-v_l) - F_l\right]d\sigma, \end{aligned} \tag{A.10}$$

where $A_{vi+1/2,j,k} = (A_{vi,j,k}A_{vi+1,j,k})/2$, $h_{i+1/2,j,k+1/2} = (h_{i,j,k} + h_{i+1,j,k} + h_{i,j,k+1} + h_{i+1,j,k+1})/4$, and functions F_1, F_k, and F_l are, respectively, defined as

$$F_1 = q_{a1}\cdot(\text{A.1a}) + u_{a1}q_1\cdot\left[(\text{A.1b}) + \frac{\tau_{1\lambda}}{q_1}\right] + v_{a1}q_1\cdot\left[(\text{A.1c}) + \frac{\tau_{1\varphi}}{q_1}\right], \tag{A.11}$$

$$F_k = q_{ak} \cdot (\text{A.2a}) + u_{ak} q_k \cdot \left[(\text{A.2b}) - \frac{\tau_{(k-1)\lambda} - \tau_{k\lambda}}{q_1} \right] + v_{ak} q_k \cdot \left[(\text{A.2c}) - \frac{\tau_{(k-1)\varphi} - \tau_{k\varphi}}{q_k} \right], \tag{A.12}$$

$$F_l = q_{al} \cdot (\text{A.3a}) + u_{al} q_l \cdot \left[(\text{A.3b}) - \frac{\tau_{(l-1)\lambda}}{q_l} \right] + v_{al} q_l \cdot \left[(\text{A.3c}) - \frac{\tau_{(l-1)\varphi}}{q_l} \right]. \tag{A.13}$$

Note that (A.11)~(A.13) do not contain the variable A_v, which means

$$\frac{\partial F_1}{\partial A_{vi,j,k}} = \frac{\partial F_k}{\partial A_{vi,j,k}} = \frac{\partial F_l}{\partial A_{vi,j,k}} = 0. \tag{A.14}$$

The Lagrangian function can be written as

$$\begin{aligned} & L(q, u, v; q_a, u_a, v_a; \mathbf{p}) \\ & = J(q, u, v; \mathbf{p}) \\ & - \int \left[\frac{u_{a1,j,1} A_{vi+1/2,j,1} \rho_{1+1/2}}{h_{i+1/2,j,1+1/2}} \left(u_{i,j,1} - u_{i,j,2} \right) + \frac{u_{a1,j,1} A_{vi+1/2,j,1} \rho_{1+1/2}}{h_{i+1/2,j,1+1/2}} \left(v_{i,j,1} - v_{i,j,2} \right) - F_1 \right] d\sigma \\ & + \cdots \\ & - \int \left[\frac{u_{ai,j,k} A_{vi+1/2,j,k} \rho_{k+1/2}}{h_{i+1/2,j,k+1/2}} \left(u_{i,j,k} - u_{i,j,k+1} \right) - \frac{u_{ai,j,k} A_{vi+1/2,j,k-1} \rho_{k-1/2}}{h_{i+1/2,j,k-1/2}} \left(u_{i,j,k-1} - u_{i,j,k} \right) \right] d\sigma \\ & - \int \left[\frac{v_{ai,j,k} A_{vi+1/2,j,k} \rho_{k+1/2}}{h_{i+1/2,j,k+1/2}} \left(v_{i,j,k} - v_{i,j,k+1} \right) - \frac{v_{ak} A_{vi+1/2,j,k-1} \rho_{k-1/2}}{h_{i+1/2,j,k-1/2}} \left(v_{i,j,k-1} - v_{i,j,k} \right) - F_k \right] d\sigma \\ & + \cdots \\ & + \int \left[\frac{u_{ai,j,l} A_{vi+1/2,j,l-1} \rho_{l-1/2}}{h_{i+1/2,j,l-1/2}} \left(u_{i,j,l-1} - u_{i,j,l} \right) + \frac{u_{ai,j,l} A_{vi+1/2,j,l-1} \rho_{l-1/2}}{h_{i+1/2,j,l-1/2}} \left(v_{i,j,l-1} - v_{i,j,l} \right) - F_l \right] d\sigma. \end{aligned} \tag{A.15}$$

Finally, according to the typical theory of Lagrangian multiplier method, we have the following first-order derivate of Lagrangian function with respect to the control parameter A_v:

$$\frac{\partial L}{\partial A_{vi,j,k}} = 0, \tag{A.16}$$

then the gradient of the cost function with respect to the variable A_v can be deduced from (A.16):

$$\begin{aligned} & \frac{\partial J}{\partial A_{vi,j,k}} \\ & = \sum_n \left[u^n_{ai,j,k} \frac{\rho_{k+1/2}}{h_{i+1/2,j,k+1/2}} \left(u^n_{i,j,k} - u^n_{i,j,k+1} \right) + u^n_{ai-1,j,k} \frac{\rho_{k+1/2}}{h_{i-1/2,j,k+1/2}} \left(u^n_{i-1,j,k} - u^n_{i-1,j,k+1} \right) \right] \\ & - \sum_n \left[u^n_{ai,j,k+1} \frac{\rho_{k+1/2}}{h_{i+1/2,j,k+1/2}} \left(u^n_{i,j,k} - u^n_{i,j,k+1} \right) + u^n_{ai-1,j,k+1} \frac{\rho_{k+1/2}}{h_{i-1/2,j,k+1/2}} \left(u^n_{i-1,j,k} - u^n_{i-1,j,k+1} \right) \right] \\ & + \sum_n \left[v^n_{ai,j,k} \frac{\rho_{k+1/2}}{h_{i+1/2,j,k+1/2}} \left(v^n_{i,j,k} - v^n_{i,j,k+1} \right) + v^n_{ai-1,j,k} \frac{\rho_{k+1/2}}{h_{i-1/2,j,k+1/2}} \left(v^n_{i-1,j,k} - v^n_{i-1,j,k+1} \right) \right] \\ & - \sum_n \left[v^n_{ai,j,k+1} \frac{\rho_{k+1/2}}{h_{i+1/2,j,k+1/2}} \left(v^n_{i,j,k} - v^n_{i,j,k+1} \right) + v^n_{ai-1,j,k+1} \frac{\rho_{k+1/2}}{h_{i-1/2,j,k+1/2}} \left(v^n_{i-1,j,k} - v^n_{i-1,j,k+1} \right) \right] \\ & = (\rho_k + \rho_{k+1}) \times \sum_n \left[\frac{\left(u^n_{i,j,k} - u^n_{i,j,k+1} \right) \left(u^n_{ai,j,k} - u^n_{ai,j,k+1} \right)}{h_{i,j,k} + h_{i+1,j,k} + h_{i,j,k+1} + h_{i+1,j,k+1}} + \frac{\left(u^n_{i-1,j,k} - u^n_{i-1,j,k+1} \right) \left(u^n_{ai-1,j,k} - u^n_{ai-1,j,k+1} \right)}{h_{i-1,j,k} + h_{i,j,k} + h_{i-1,j,k+1} + h_{i,j,k+1}} \right] \\ & + (\rho_k + \rho_{k+1}) \times \sum_n \left[\frac{\left(v^n_{i,j,k} - v^n_{i,j,k+1} \right) \left(v^n_{ai,j,k} - v^n_{ai,j,k+1} \right)}{h_{i,j,k} + h_{i+1,j,k} + h_{i,j,k+1} + h_{i+1,j,k+1}} + \frac{\left(v^n_{i-1,j,k} - v^n_{i-1,j,k+1} \right) \left(v^n_{ai-1,j,k} - v^n_{ai-1,j,k+1} \right)}{h_{i-1,j,k} + h_{i,j,k} + h_{i-1,j,k+1} + h_{i,j,k+1}} \right]. \end{aligned} \tag{A.17}$$

Conflict of Interests

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

Acknowledgments

Partial support for this research was provided by the National Natural Science Foundation of China through Grant 41371496, the State Ministry of Science and Technology of China through Grant 2013AA122803, and the Fundamental Research Funds for the Central Universities 201262007 and 201362033.

References

[1] M. Rattray, "On the coastal generation of internal tides," *Tellus*, vol. 12, no. 1, pp. 54–62, 1960.

[2] P. G. Baines, "The generation of internal tides by flat-bump topography," *Deep-Sea Research and Oceanographic Abstracts*, vol. 20, no. 2, pp. 179–205, 1973.

[3] T. H. Bell, "Topographically generated internal waves in the open ocean," *Journal of Geophysical Research*, vol. 80, pp. 320–327, 1975.

[4] P. G. Baines, "On internal tide generation models," *Deep Sea Research A: Oceanographic Research Papers*, vol. 29, no. 3, pp. 307–338, 1982.

[5] P. D. Craig, "Solutions for internal tidal generation over coastal topography," *Journal of Marine Research*, vol. 45, pp. 83–105, 1987.

[6] T. Gerkema, "A unified model for the generation and fission of internal tides in a rotating ocean," *Journal of Marine Research*, vol. 54, no. 3, pp. 421–450, 1996.

[7] S. G. Llewellyn Smith and W. R. Young, "Conversion of the barotropic tide," *Journal of Physical Oceanography*, vol. 32, no. 5, pp. 1554–1566, 2002.

[8] S. K. Kang, M. G. G. Foreman, W. R. Crawford, and J. Y. Cherniawsky, "Numerical modeling of internal tide generation along the Hawaiian Ridge," *Journal of Physical Oceanography*, vol. 30, no. 5, pp. 1083–1098, 2000.

[9] Y. Niwa and T. Hibiya, "Numerical study of the spatial distribution of the M_2 internal tide in the Pacific Ocean," *Journal of Geophysical Research C: Oceans*, vol. 106, no. 10, pp. 22441–22449, 2001.

[10] P. F. Cummins, J. Y. Cherniawsky, and M. G. G. Foreman, "North Pacific internal tides from the Aleutian ridge: altimeter observations and modeling," *Journal of Marine Research*, vol. 59, no. 2, pp. 167–191, 2001.

[11] Y. Niwa and T. Hibiya, "Three-dimensional numerical simulation of M_2 internal tides in the East China Sea," *Journal of Geophysical Research C: Oceans*, vol. 109, no. 4, Article ID C04027, 2004.

[12] S. Jan, C.-S. Chern, J. Wang, and S.-Y. Chao, "Generation of diurnal K1 internal tide in the Luzon Strait and its influence on surface tide in the South China Sea," *Journal of Geophysical Research C: Oceans*, vol. 112, no. 6, Article ID C06019, 2007.

[13] R. C. Pacanowski and S. G. H. Philander, "Parameterization of vertical mixing in numerical models of tropical oceans," *Journal of Physical Oceanography*, vol. 11, no. 11, pp. 1443–1451, 1981.

[14] B. Henderson-Sellers, "A simple formula for vertical eddy diffusion coefficients under conditions of nonneutral stability," *Journal of Geophysical Research*, vol. 87, pp. 5860–5864, 1982.

[15] N. S. Heaps, "Three-dimensional model for tides and surges with vertical eddy viscosity prescribed in two layers—I. Mathematical formulation," *Geophysical Journal of the Royal Astronomical Society*, vol. 64, pp. 291–302, 1981.

[16] N. S. Heapsand and J. E. Jones, "Three-dimensional model for tides and surges with vertical eddy viscosity prescribed in two layers—II. Irish Sea with bed friction layer," *Geophysical Journal of the Royal Astronomical Society*, vol. 64, pp. 303–320, 1981.

[17] V. P. Kochergin, "Three-dimensional prognostic models. In: three-dimensional coastal ocean models," *Coastal and Estuarine Science*, vol. 4, pp. 201–208, 1987.

[18] T. Pohlmann, "Calculating the annual cycle of the vertical eddy viscosity in the North Sea with a three-dimensional baroclinic shelf sea circulation model," *Continental Shelf Research*, vol. 16, no. 2, pp. 147–161, 1996.

[19] A. F. Bennett and P. C. McIntosh, "Open ocean modeling as an inverse problem: tidal theory," *Journal of Physical Oceanography*, vol. 12, no. 10, pp. 1004–1018, 1982.

[20] L. Yu and J. J. O'Brien, "Variational estimation of the wind stress drag coefficient and the oceanic eddy viscosity profile," *Journal of Physical Oceanography*, vol. 21, pp. 709–719, 1991.

[21] R. W. Lardner, "Optimal control of open boundary conditions for a numerical tidal model," *Computer Methods in Applied Mechanics and Engineering*, vol. 102, no. 3, pp. 367–387, 1993.

[22] U. Seiler, "Estimation of open boundary conditions with the adjoint method," *Journal of Geophysical Research*, vol. 98, no. 12, pp. 22855–22870, 1993.

[23] I. M. Navon, "Practical and theoretical aspects of adjoint parameter estimation and identifiability in meteorology and oceanography," *Dynamics of Atmospheres and Oceans*, vol. 27, no. 1–4, pp. 55–79, 1998.

[24] N. Ayoub, "Estimation of boundary values in a North Atlantic circulation model using an adjoint method," *Ocean Modelling*, vol. 12, no. 3-4, pp. 319–347, 2006.

[25] J. Zhang and X. Lu, "Inversion of three-dimensional tidal currents in marginal seas by assimilating satellite altimetry," *Computer Methods in Applied Mechanics and Engineering*, vol. 199, no. 49–52, pp. 3125–3136, 2010.

[26] J. C. Zhang and X. Q. Lu, "Parameter estimation for a three-dimensional numerical barotropic tidal model with adjoint method," *International Journal for Numerical Methods in Fluids*, vol. 57, no. 1, pp. 47–92, 2008.

[27] H. Chen, C. Miao, and X. Lv, "A three-dimensional numerical internal tidal model involving adjoint method," *International Journal for Numerical Methods in Fluids*, vol. 69, no. 10, pp. 1584–1613, 2012.

[28] J. Zhang and H. Chen, "Semi-idealized study on estimation of partly and fully space varying open boundary conditions for tidal models," *Abstract and Applied Analysis*, vol. 2013, Article ID 282593, 14 pages, 2013.

[29] A. Cao, H. Chen, J. Zhang, and X. Lv, "Optimization of open boundary conditions in a 3D internal tidal model with the adjoint method around Hawaii," *Abstract and Applied Analysis*, vol. 2013, Article ID 950926, 11 pages, 2013.

[30] H. Chen, C. Miao, and X. Lv, "Estimation of open boundary conditions for an internal tidal model with adjoint method: a comparative study on optimization methods," *Mathematical Problems in Engineering*, vol. 2013, Article ID 802136, 12 pages, 2013.

[31] X. Lu and J. Zhang, "Numerical study on spatially varying bottom friction coefficient of a 2D tidal model with adjoint method," *Continental Shelf Research*, vol. 26, no. 16, pp. 1905–1923, 2006.

[32] J. Zhang, X. Lu, P. Wang, and Y. P. Wang, "Study on linear and nonlinear bottom friction parameterizations for regional tidal

models using data assimilation," *Continental Shelf Research*, vol. 31, no. 6, pp. 555–573, 2011.

[33] R. W. Lardner and S. K. Das, "Optimal estimation of eddy viscosity for a quasi-three-dimensional numerical tidal and storm surge model," *International Journal for Numerical Methods in Fluids*, vol. 18, no. 3, pp. 295–312, 1994.

[34] J. E. Richardson and V. G. Panchang, "A modified adjoint method for inverse eddy viscosity estimation for use in coastal circulation models," in *Proceedings of the 2nd International Conference on Estuarine and Coastal Modeling*, pp. 733–745, November 1992.

[35] G. P. Cressman, "An operational objective analysis system," *Monthly Weather Review*, vol. 87, no. 10, pp. 367–374, 1959.

[36] S. J. Wright and J. Nocedal, *Numerical optimization*, Springer, New York, NY, USA, 1999.

[37] W. F. Mascarenhas, "The BFGS method with exact line searches fails for non-convex objective functions," *Mathematical Programming*, vol. 99, no. 1, pp. 49–61, 2004.

[38] A. K. Alekseev, I. M. Navon, and J. L. Steward, "Comparison of advanced large-scale minimization algorithms for the solution of inverse ill-posed problems," *Optimization Methods and Software*, vol. 24, no. 1, pp. 63–87, 2009.

[39] X. Zou, I. M. Navon, M. Berger, K. H. Phua, T. Schlick, and F. Le Dimet, "Numerical experience with limited-memory quasi-Newton and truncated Newton methods," *SIAM Journal on Optimization*, vol. 3, no. 3, pp. 582–608, 1993.

[40] J. Nocedal, "Updating quasi-Newton matrices with limited storage," *Mathematics of Computation*, vol. 35, no. 151, pp. 773–782, 1980.

[41] R. Malouf, "A comparison of algorithms for maximum entropy parameter estimation," in *Proceedings of the 6th Conference on Natural Language Learning (CoNLL '02)*, pp. 49–55, 2002.

[42] G. Andrew and J. Gao, "Scalable training of L1-regularized log-linear models," in *Proceedings of the 24th International Conference on Machine Learning (ICML '07)*, pp. 33–40, June 2007.

[43] A. K. Alekseev, I. M. Navon, and J. L. Steward, "Comparison of advanced large-scale minimization algorithms for the solution of inverse ill-posed problems," *Optimization Methods & Software*, vol. 24, no. 1, pp. 63–87, 2009.

[44] D. C. Liu and J. Nocedal, "On the limited memory BFGS method for large scale optimization," *Mathematical Programming*, vol. 45, no. 1–3, pp. 503–528, 1989.

[45] L. J. Nocedal, "BFGS subroutine, software for large-scale unconstrained optimization," http://www.ece.northwestern.edu/~nocedal/lbfgs.html.

Hindawi Publishing Corporation
Mathematical Problems in Engineering
Volume 2015, Article ID 414727, 8 pages
http://dx.doi.org/10.1155/2015/414727

Research Article

A Directly Numerical Algorithm for a Backward Time-Fractional Diffusion Equation Based on the Finite Element Method

Zhousheng Ruan,[1,2,3] Zewen Wang,[2] and Wen Zhang[1,2]

[1] *Fundamental Science on Radioactive Geology and Exploration Technology Laboratory, East China Institute of Technology, Nanchang, Jiangxi 330013, China*
[2] *School of Science, East China Institute of Technology, Nanchang, Jiangxi 330013, China*
[3] *School of Mathematics and Statistics, Wuhan University, Wuhan, Hubei 430072, China*

Correspondence should be addressed to Zewen Wang; zwwang6@gmail.com

Received 8 April 2014; Accepted 22 July 2014

Academic Editor: Valery G. Yakhno

We study a backward problem for a time-fractional diffusion equation, which is formulated into a regularized optimization problem. After solving a sequence of well-posed direct problems by the finite element method, a directly numerical algorithm is proposed for solving the regularized optimization problem. In order to obtain a reasonable regularization solution, we utilize the discrepancy principle with decreasing geometric sequence to choose regularization parameters. One- and two-dimensional examples are given to verify the efficiency and stability of the proposed method.

1. Introduction

Nowadays, there is increasing attention on fractional diffusion equations which can be used to describe anomalous diffusion phenomena instead of classical diffusion process. These new fractional-order models are more efficient than the integer-order models, because the fractional-order derivatives and integrals enable the description of the memory and hereditary properties of different substance [1]. By an argument similar to the derivation of the classical diffusion equation from Brownian motion, one can derive a fractional diffusion equation from continuous-time random walk. For example, in paper [2] the authors illustrated a fractional diffusion with respect to a non-Markovian diffusion process, while the authors discussed continuous-time random walks on fractals in paper [3].

We notice that mathematical and numerical analysis of the direct problems of the time-fractional diffusion equations has aroused wide concern in recent years; see [4–10] and references therein. At the same time, the inverse problems for the time-fractional diffusion equations have attracted more and more attention, not only for theoretical analysis but also for popular applications. The authors concluded that there exists a unique weak solution for the backward time-fractional diffusion equation problem under the overdetermined condition $u(x,T) \in H^2(\Omega) \cap H_0^1(\Omega)$ in paper [4]. The authors of papers [11–13] considered the backward problem of the time-fractional diffusion equation and proposed, respectively, a quasi-reversibility method, an optimization method, and a data regularization method for reconstructing the initial value. Inverse source problems for time-fractional diffusion equations were studied by using the method of the eigenfunction expansion [14], the integral equation method [15], and the separation of variables method [16], respectively, for recovering the space-dependent or time-dependent source term. In [17], the authors recovered the temperature function from one measured temperature at one interior point of a one-dimensional semi-infinite fractional diffusion equation based on Dirichlet kernel mollification techniques. The authors studied an inverse problem of identifying a spatially varying potential term in a one-dimensional time-fractional diffusion equation from the flux measurements taken at a single fixed time corresponding to a given set of input sources in [18]. Recently, for determining the space-dependent source in a parabolic equation, the authors [19] proposed a regularized optimization method

together with the linear model function method [19, 20] for choosing regularization parameters. Inspired by this noniterative optimization method, we develop it to solve the backward problem for a time-fractional diffusion equation in this paper.

Let α be a constant such that $0 < \alpha < 1$. We consider the following time-fractional diffusion equation:

$$\frac{\partial^\alpha u(x,t)}{\partial t^\alpha} = (Lu)(x,t), \quad (x,t) \in \Omega \times (0,T) \tag{1}$$

with homogeneous boundary condition

$$u(x,t)|_{\partial\Omega} = 0, \quad t \in [0,T] \tag{2}$$

and initial condition

$$u(x,t)|_{t=0} = \varphi(x), \quad x \in \overline{\Omega}, \tag{3}$$

where Ω is a bounded domain in R^d ($d \geq 1$) and L is symmetric uniformly elliptic operator given by

$$L(u) = \sum_{i=1}^{d} \frac{\partial}{\partial x_i} \left(\sum_{j=1}^{d} \theta_{i,j} \frac{\partial}{\partial x_j} u(x) \right) - c(x)\, u(x); \tag{4}$$

that is, there exists a constant $\nu > 0$, such that $\nu \sum_i^d \xi_i^2 \leq \sum_{i,j=1}^d \theta_{i,j}\xi_i\xi_j$, $x \in \overline{\Omega}$, and $\xi \in \mathbb{R}^d$. The coefficients satisfy

$$\begin{gathered} \theta_{i,j} \in C^1\left(\overline{\Omega}\right), \quad \theta_{i,j} = \theta_{j,i}, \\ c(x) \in C\left(\overline{\Omega}\right), \quad c(x) \geq 0, \ \forall x \in \overline{\Omega}. \end{gathered} \tag{5}$$

Here, $\partial^\alpha u(x,t)/\partial t^\alpha$ is the Caputo fractional derivative which is defined by

$$\begin{aligned} &\frac{\partial^\alpha u(x,t)}{\partial t^\alpha} \\ &\quad = \frac{1}{\Gamma(1-\alpha)} \int_0^t (t-\eta)^{-\alpha} \frac{\partial u}{\partial \eta} \mathrm{d}\eta, \quad 0 < \alpha < 1, \end{aligned} \tag{6}$$

where $\Gamma(1-\alpha)$ is the Gamma function.

If the function $\varphi(x)$ and the coefficients in (1) are all known, problem (1)–(3) is the so-called direct problem that can be solved stably by the finite element method, the finite difference, the spectrum method, and so forth. Here, we focus on the backward problem; that is, we try to determine the initial value $\varphi(x)$ by the additional data $g(x)$ which is the measurement of the exact value $u(x,T)$ and satisfies

$$\left\| u(\cdot,T) - g(\cdot) \right\|_{L^2(\Omega)} \leq \delta \tag{7}$$

for some known error level $\delta > 0$. As we all know, the backward problem is ill-posed, which means that the solution does not depend continuously on the given data and any small perturbation in the given data may cause large change to the solution. For overcoming the ill-posedness we will adopt Tikhonov regularization in our treatment.

The rest of the paper is organized as follows. In Section 2, we reformulate the direct problem in a weak and variational sense. Then we formulate the inverse problem into a regularized optimization problem in Section 3. In Section 4, we give implementations of the regularized optimization method. Finally, numerical results are given to illustrate the efficiency and stability of the proposed method.

2. Weak Form and Weak Solution

The weak form of problem (1)–(3) is finding $u(\cdot,t) \in H_0^1(\Omega)$ such that

$$\begin{gathered} \left\langle \frac{\partial^\alpha u}{\partial t^\alpha}, \chi \right\rangle + a(u,\chi) = 0, \quad \forall \chi \in X,\ 0 < t < T, \\ u(0) = \varphi(x), \end{gathered} \tag{8}$$

where $X = H_0^1(\Omega)$, $\langle \cdot,\cdot \rangle$ denotes the inner product in $L^2(\Omega)$, and

$$a(u,\chi) = \int_\Omega \left(\sum_{i,j=1}^{d} \theta_{i,j} \frac{\partial u(x)}{\partial x_i} \cdot \chi_{x_j} - c(x)\, u \cdot \chi \right) \mathrm{d}x. \tag{9}$$

Definition 1. A function $u(x,t)$ is said to be a weak solution of the direct problem (1)–(3) if $u \in C([0,T]; L^2(\Omega)) \cap C((0,T]; H^2(\Omega) \cap H_0^1(\Omega))$ and the weak form (8) is satisfied.

Lemma 2 (see [4]). *If $0 < \alpha < 1$, $\varphi \in L^2(\Omega)$, there exists a unique weak solution $u \in C([0,T]; L^2(\Omega)) \cap C((0,T]; H^2(\Omega) \cap H_0^1(\Omega))$ to problem (1)–(3), and the expression of the weak solution can be formulated by the following eigenfunction expansion:*

$$u(x,t;\varphi) = \sum_{k=1}^{\infty} E_{\alpha,1}\left(-\lambda_k t^\alpha\right) (\varphi, \chi_k)\, \chi_k, \tag{10}$$

where $E_{\alpha,\gamma}(z)$ is the double-parameter Mittag-Leffler function and is defined by

$$E_{\alpha,\gamma}(z) := \sum_{k=0}^{\infty} \frac{z^k}{\Gamma(\alpha k + \gamma)}, \quad z \in \mathbb{C},\ \alpha > 0,\ \gamma \geq 0; \tag{11}$$

$\{\lambda_i\}_{j=1}^\infty$ ($0 < \lambda_1 \leq \lambda_2 \leq \cdots, i = 1, \ldots, \infty$) and $\{\chi_i\}_{j=1}^\infty$ are the Dirichlet eigenvalues and the orthonormal eigenfunctions of symmetric uniformly elliptic operator $-L$, respectively.

The following two propositions will be used in the context.

Proposition 3. *$E_{\alpha,1}(-t)$ is a completely monotonic decreasing function for $t > 0$ and satisfies*

$$1 = \left|E_{\alpha,1}(0)\right| > \left|E_{\alpha,1}\left(-\lambda_1 t^\alpha\right)\right| > \left|E_{\alpha,1}\left(-\lambda_2 t^\alpha\right)\right| > \cdots > 0. \tag{12}$$

Proposition 4. *Let $0 < \alpha < 2$, $\beta \in \mathbb{R}$, and $\pi\alpha/2 < \mu < \min\{\pi, \pi\alpha\}$. Then there exists a constant $C = C(\alpha,\beta,\mu) > 0$ such that*

$$\left|E_{\alpha,\beta}(z)\right| \leq \frac{C}{1+|z|}, \quad \mu \leq \left|\arg(z)\right| \leq \pi, \quad z \in \mathbb{C}. \tag{13}$$

3. The Regularized Optimization Problem

In this section, we will propose a regularized optimization method together with its implementations for solving the considered backward problem.

3.1. Regularized Optimization Functional. From results of Lemma 2, formula (10) gives a uniquely weak solution $u(x,t;\varphi)$ for any initial value $\varphi(x) \in L^2(\Omega)$. Naturally, it defines a forward operator

$$M : \varphi(x) \longmapsto u(x,T;\varphi). \tag{14}$$

Clearly, the forward operator M is a linear map and has the following property.

Lemma 5. *The operator M is a well-defined bounded linear operator from $L^2(\Omega)$ to $L^2(\Omega)$. Moreover, it is injective and compact.*

Proof. From Lemma 2, the solution $u(x,t;\varphi)$ can be represented by

$$u(x,t;\varphi) = \sum_{k=1}^{\infty} E_{\alpha,1}\left(-\lambda_k t^{\alpha}\right)(\varphi,\chi_k)\chi_k. \tag{15}$$

From Proposition 3, we know that $0 \le E_{\alpha,1}(-t) < 1$ for $t \in [\sigma,+\infty)$; σ is a very small positive number. By the orthogonality of $\{\chi_i\}_{j=1}^{\infty}$ and Proposition 4, we obtain

$$\begin{aligned}\|\Delta u(\cdot,t)\|^2_{L^2(\Omega)} &= \sum_{k=1}^{\infty}\lambda_k^2 E^2_{\alpha,1}\left(-\lambda_k t^{\alpha}\right)^2\langle\varphi,\chi_k\rangle^2\\ &\le \sum_{k=1}^{\infty}\frac{C^2}{\sigma^{2\alpha}}\langle\varphi,\chi_k\rangle^2\\ &= \frac{C^2}{\sigma^{2\alpha}}\|\varphi\|^2_{L^2(\Omega)}, \quad \forall t\in[\sigma,+\infty),\end{aligned} \tag{16}$$

which implies that $\|u\|_{H^2(\Omega)} \le C\|\varphi\|_{L^2(\Omega)}$. Then by *Sobolev* embedding theorem, we conclude the compactness of the operator M. □

Results of Lemma 5 show that the backward problem is ill-posed due to the compactness of operator M. Thus, regularization is necessary for recovering the initial value $\varphi(x)$. To this end, we consider a Tikhonov functional as

$$J(\varphi) = \frac{1}{2}\|u(x,T;\varphi)-g\|^2_{L^2(\Omega)} + \frac{\beta}{2}\|\varphi\|^2_{L^2(\Omega)}, \tag{17}$$

where $\varphi(x) \in L^2(\Omega)$ and β is a regularization parameter balancing the fidelity term and the smoothness of the solution. Due to the L^2-regularization term $(\beta/2)\|\varphi\|^2_{L^2(\Omega)}$, the cost functional $J(\varphi)$ is strongly convex. Subsequently, the unique existence of the minimizer can be obtained by standard arguments.

Theorem 6. *There exists a unique minimizer $\varphi^{\star}$ to $J(\varphi)$ for any given $\beta > 0$.*

Now, we formulate the backward problem into the following minimization problem:

$$\min_{\varphi\in L^2(\Omega)} J(\varphi). \tag{18}$$

3.2. Finite Element Method Approximation. Obviously, problem (18) is a function space minimization problem. Here, we use the finite element method to approximate it. Similar to that done in [19, 21], we first triangulate the domain Ω with a regular triangulation T_h of simplicial elements; let $\{p_i\}_{i=0}^{N_1}$ be the set of the nodes, and define V_h to be the continuous piecewise linear finite element space defined over T_h; that is,

$$V_h = \left\{v : v\in C_0(\Omega), v|_{\Delta_h}\in P_1(\Delta_h), \forall\Delta_h\in T_h\right\}. \tag{19}$$

Then any $u_h \in V_h$ can be repeated as $u_h = \sum_{i=0}^{N_1} u_i\psi_i$, where u_i is the value of $u_h(x)$ at point p_i, and ψ_i is the pyramid function; that is,

$$\psi_i(p_j) = \begin{cases}1, & i = j\\ 0, & i\ne j.\end{cases} \tag{20}$$

Next, we need to consider the discretization of the bounded linear operator M. We will adopt the discrete *Galerkin* method to solve the direct problem (1)–(3). The time interval $[0,T]$ is partitioned into N_2 equal subintervals by using nodal points $0 = t_0 < t_1 < \cdots < t_{N_2-1} < t_{N_2} = T$, with $t_k = k\tau$, $\tau = T/N_2$. Then, the time-fractional derivative $\partial^{\alpha}u(x,t)/\partial t^{\alpha}$ at t_k is estimated by

$$\begin{aligned}&\left.\frac{\partial^{\alpha}u(x,t)}{\partial t^{\alpha}}\right|_{t=t_k}\\ &\quad= \frac{1}{\Gamma(1-\alpha)}\int_0^{t_k}(t_k-\eta)^{-\alpha}\frac{\partial u(x,\eta)}{\partial\eta}\mathrm{d}\eta\\ &\quad= \frac{1}{\Gamma(1-\alpha)}\sum_{l=1}^{k}\int_{t_{l-1}}^{t_l}(t_k-\eta)^{-\alpha}\frac{\partial u(x,\eta)}{\partial\eta}\mathrm{d}\eta\\ &\quad\approx \frac{\tau^{-\alpha}}{\Gamma(2-\alpha)}\sum_{l=1}^{k}\left(u(x,t_l)-u(x,t_{l-1})\right)\\ &\qquad\times\left((k+1-l)^{1-\alpha}-(k-l)^{1-\alpha}\right)\\ &\quad= \frac{\tau^{-\alpha}}{\Gamma(2-\alpha)}\sum_{l=1}^{k}\omega_l\left(u(x,t_{k+1-l})-u(x,t_{k-l})\right),\end{aligned} \tag{21}$$

where $\omega_l = l^{1-\alpha}-(l-1)^{1-\alpha}$, $l = 1,2,\ldots,k$, $k = 1,2,\ldots,N_2$. Denote by $u_h^k \in V_h$ the approximation of $u(\cdot,t_k)$ and

$$Du^{\alpha,k}_{h,t} = \frac{\tau^{-\alpha}}{\Gamma(2-\alpha)}\sum_{l=1}^{k}\omega_l\left(u_h^{k+1-l}-u_h^{k-l}\right). \tag{22}$$

Now we define the fully discrete finite element method by

$$\left\langle Du^{\alpha,k}_{h,t},\psi\right\rangle + a\left(u_h^k,\psi\right) = 0, \qquad u^0 = \varphi(x) \tag{23}$$

for any $\psi\in\mathring{V}_h$, where $a(u_h^k,\psi) = \int_{\Omega}(\sum_{i,j=1}^{d}\theta_{i,j}(\partial u_h^k/\partial x_i)\cdot\psi_{x_j} - c(x)u_h^k\cdot\psi)\mathrm{d}x$. The space $\mathring{V}_h$, in which all functions vanish on the boundary $\partial\Omega$, is a subspace of V_h. Clearly, (23) is a linear system about u_h^k, $k = 1,2,\ldots,N_2$. Subsequently, there exists a discrete linear operator M_h such that

$$M_h : \varphi \longmapsto u^k, \quad k = 1,2,\ldots,N_2. \tag{24}$$

Theorem 7. *Let u and u_h^k be the weak solution of (1)–(3) and the discrete Galerkin finite element solution of (23), respectively. Then there is a constant $C > 0$ such that, for $0 < \alpha < 1$,*

$$\left\|u(\cdot, t_k) - u_h^k\right\|_{L^2(\Omega)} \le C\left(\tau^{2-\alpha} + h\right), \quad k = 1, 2, \ldots, N_2, \tag{25}$$

where C is independent of h, α, and τ.

The proof of Theorem 7 follows the same lines as the proof of Theorem 2.1 in [22]. So, we omit it.

3.3. Implementations of the Regularized Optimization Method. Applying the interpolation of finite element, the initial value function $\varphi(x)$ can be written approximately in the finite element form of

$$\varphi \approx \varphi_h = \sum_{i=0}^{N_1} \varphi_i \psi_i(x), \tag{26}$$

where $\varphi_i := \varphi(x_i)$. Due to the linearity of the homogeneous governing equation and the homogeneous boundary condition, we easily see that problem (1)–(3) satisfies the principle of superposition. Here, we also use this principle of superposition to formulate the continuous problem (18) into the following discrete problem:

$$\begin{aligned} \min_{\varphi_h \in V_h} J(\varphi_h) = {} & \frac{1}{2}\int_\Omega \left(\sum_{i=0}^{N_1} \varphi_i u_{h,i}^{N_2} - g_h\right)^2 \mathrm{d}x \\ & + \frac{\beta}{2}\int_\Omega \left(\sum_{i=0}^{N_1} \varphi_i \psi_i(x)\right)^2 \mathrm{d}x, \end{aligned} \tag{27}$$

where $u_{h,i}^k$, $k = 1, 2, \ldots, N_2$, is the finite element solution of $u(x, t)$ and satisfies

$$\begin{aligned} u_{h,i}^0 &= \psi_i, \\ \left\langle D u_{h,i}^{\alpha,k}, \psi_j \right\rangle + a\left(u_{h,i}^k, \psi_j\right) &= 0, \end{aligned} \tag{28}$$

for any $\psi_j \in \mathring{S}_h$ and $k = 1, 2, \ldots, N_2$, where

$$D u_{h,i}^{\alpha,k} = \frac{\tau^{-\alpha}}{\Gamma(2-\alpha)} \sum_{l=1}^{k} \omega_l \left(u_{h,i}^{k+1-l} - u_{h,i}^{k-l}\right),$$

$$g_h(x) = \sum_{i=0}^{N_1} g_i \psi_i(x),$$

$$a\left(u_{h,i}^k, \psi_j\right) = \int_\Omega \left(\sum_{i,j=1}^{d} \theta_{i,j} \frac{\partial u_{h,i}^k}{\partial x_i} \cdot \psi_{j_{x_j}} - c(x)\, u_{h,i}^k \cdot \psi_j\right) \mathrm{d}x. \tag{29}$$

Therefore, numerical solving of the backward problem is essential to determine the $(N_1 + 1)$-dimensional real vector $\Phi = [\varphi_0, \ldots, \varphi_{N_1}]^T$.

From the necessary condition for minimizing the approximation function $J(\varphi_h)$, that is,

$$\frac{\partial J(\varphi_h)}{\partial \varphi_i} = 0, \quad i = 0, 1, \ldots, N_1, \tag{30}$$

we obtain a linear algebraic system

$$A\Phi = F, \tag{31}$$

$$\begin{aligned} A &= \left[\int_\Omega \left(u_{h,i}^{N_2}(x) u_{h,j}^{N_2}(x) + \beta \psi_i \psi_j\right) dx\right]_{(N_1+1)\times(N_1+1)}, \\ \Phi &= \left[\varphi_0, \varphi_1, \ldots, \varphi_{N_1}\right]^T, \\ F &= \left[\int_\Omega g_h(x) u_{h,i}^{N_2}(x) \mathrm{d}x\right]_{(N_1+1)\times 1}. \end{aligned} \tag{32}$$

Let $\Phi^* = [\varphi_0^*, \varphi_1^*, \ldots, \varphi_{N_1}^*]^T$ be the solution of (31) for a given regularization parameter β. Then, we obtain the approximation solution of φ as follows:

$$\varphi_h = \sum_{i=0}^{N_1} \varphi_i^* \psi_i(x). \tag{33}$$

4. Method for Choosing Regularization Parameters

As we all know, the backward problem for determining the initial value is an ill-posed problem; that is, the round-off errors and the measurement noises may be highly amplified due to the choice of an unreasonable regularization parameter, therefore making the regularization solution completely useless [19, 20]. Because of the important role of regularization parameters, a good strategy for selecting regularization parameters should be taken in the computational process. For a fixed $0 < r < 1$ and $\beta_0 > 0$, we consider a geometric sequence of regularization parameters

$$\beta_k = \beta_0 r^k, \quad k \in \mathbb{N}. \tag{34}$$

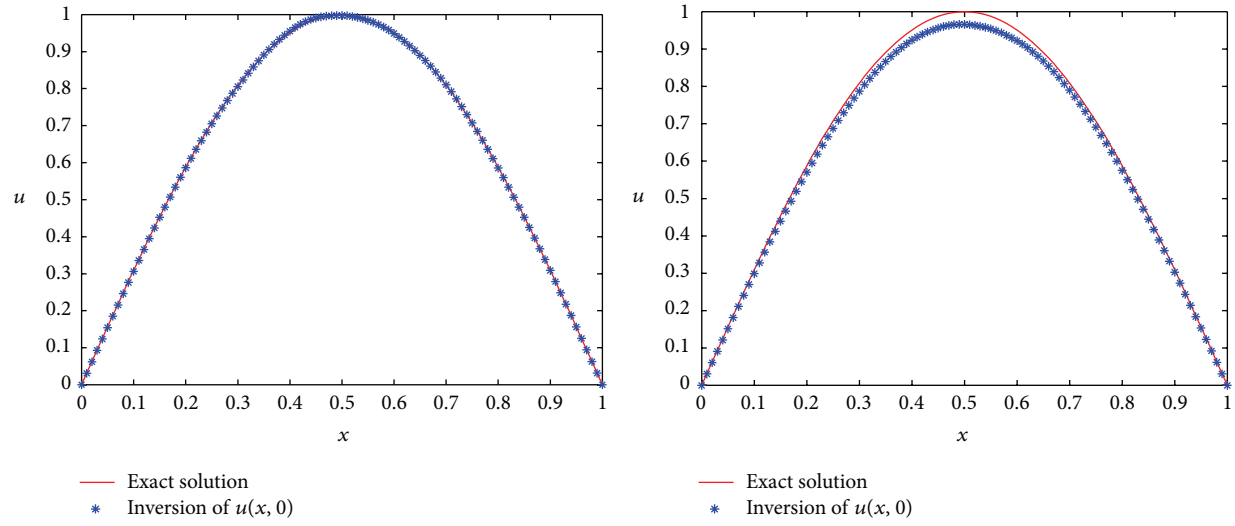
Then, we employ the discrepancy principle to choose a regularization parameter β_{k^*} after k^* steps with

$$\begin{aligned} \int_\Omega \left(u_h^{k^*}(x, T) - g_h\right)^2 \mathrm{d}x \le \delta^2 < \int_\Omega \left(u_h^k(x, T) - g_h\right)^2 \mathrm{d}x, \\ 0 \le k < k^*, \end{aligned} \tag{35}$$

where $u_h^k(x, t)$ is the finite element solution with respect to φ_h and β_k.

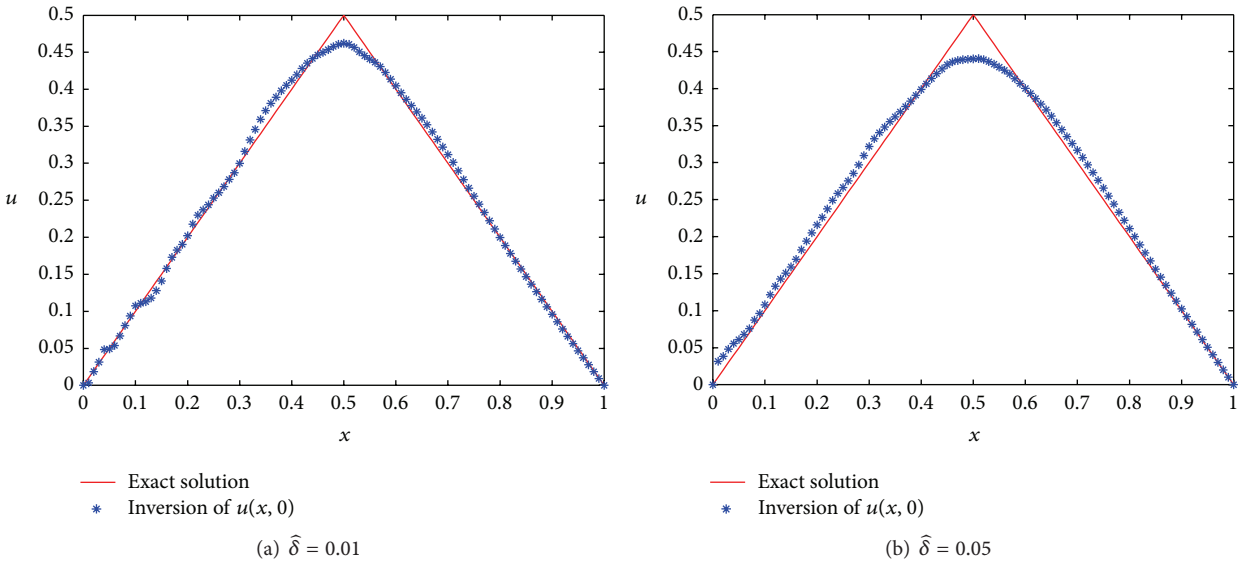
5. Numerical Examples

In all one-dimensional examples, $\Omega = [0, 1]$, we divide Ω into 100 equal subintervals which means that there are 100 elements and 101 nodes, $N_2 = 100$, $(Lu)(x, t) = (\mathrm{d}/\mathrm{d}x)(\theta(x)(\mathrm{d}/\mathrm{d}x)u) - c(x)u$. In all two-dimensional examples, $\Omega = [0, 1] \times [0, 1]$, we divide Ω into 1024 equal triangle element, $N_2 = 80$, $(Lu)(x, t) = \nabla(\theta(x, y)\nabla u) - c(x, y)u$. In the computational process, the measurement vector g_h is obtained actually at the points of the mesh grid and added by randomly distributed perturbations with relative noise level $\widehat{\delta}$; that is, $g_h = u_h(x, T) * (1 + \widehat{\delta} * (2 * \mathrm{rand}(\mathrm{size}(u_h(x, T))) - 1))$.



(a) $\widehat{\delta} = 0.01$ (b) $\widehat{\delta} = 0.05$

FIGURE 1: Comparison between exact solution and inverse solution for Example 1.



(a) $\widehat{\delta} = 0.01$ (b) $\widehat{\delta} = 0.05$

FIGURE 2: Comparison between exact solution and inverse solution for Example 2.

We take $\beta_0 = 1$ and $r = 0.1$ in all numerical examples. The relative error of the inverse solutions is defined by

$$\text{RelError} = \frac{\left\|\varphi_{\text{exact}} - \varphi_{\text{inversion}}\right\|_{L^2(\Omega)}}{\left\|\varphi_{\text{exact}}\right\|_{L^2(\Omega)}}. \tag{36}$$

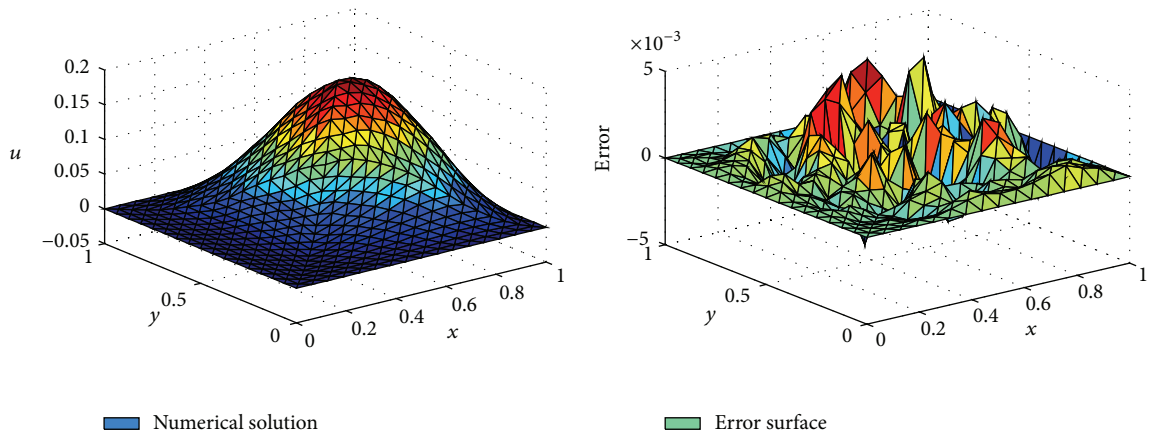
Example 1. We take $\alpha = 0.9$, $\theta(x) = 1$, and $c(x) = 0$. Let the exact initial value for problem (1)–(3) be $\sin(\pi x)$. Numerical results for relative noise levels 1% and 5% are shown and listed in Figure 1 and Table 1.

Example 2. Let the exact initial value for problem (1)–(3) be

$$\varphi(x) = \begin{cases} x, & x \in \left[0, \frac{1}{2}\right] \\ (1-x), & x \in \left[\frac{1}{2}, 1\right], \end{cases} \tag{37}$$

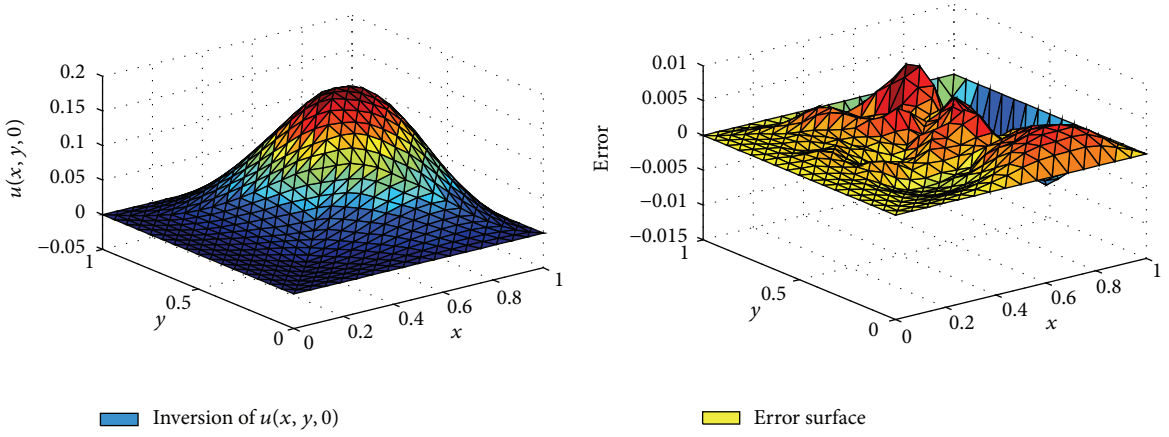
$\theta(x) = x$, $c(x) = -0.001x$, and $\alpha = 0.6$. Numerical results with the relative noise levels 1% and 5% are shown in Figure 2 and listed in Table 1.

Example 3. Let the exact initial value for problem (1)–(3) be $10x^3y^2(1-x)(1-y)$. And we take $\alpha = 0.8$, $\theta(x, y) = 1$,



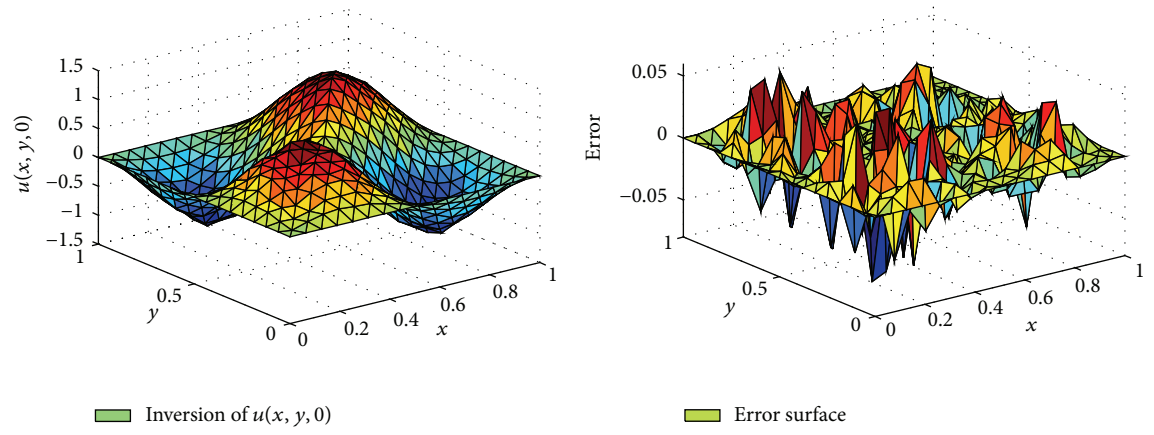
(a) Inverse solution (b) Error surface

Figure 3: Results with $\widehat{\delta} = 0.01$ for Example 3.



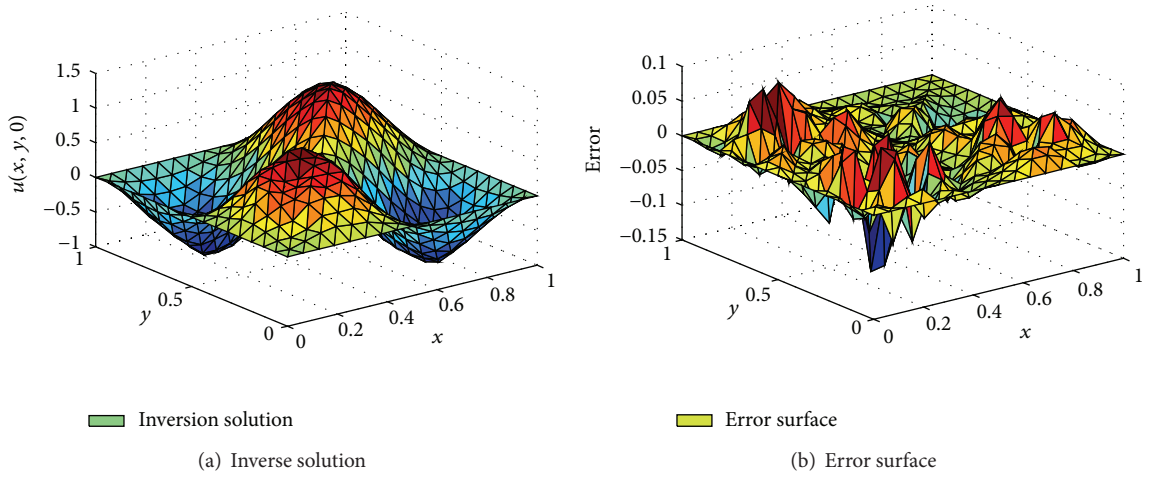
(a) Inverse solution (b) Error surface

Figure 4: Results with $\widehat{\delta} = 0.05$ for Example 3.



(a) Inverse solution (b) Error surface

Figure 5: Results with $\widehat{\delta} = 0.01$ for Example 4.



(a) Inverse solution

(b) Error surface

FIGURE 6: Results with $\widehat{\delta} = 0.05$ for Example 4.

TABLE 1: Some numerical results for Examples 1–4.

Examples	$\widehat{\delta}$	β	RelError
Example 1	0.01	$1e-6$	$3.05e-3$
	0.05	$1e-5$	$1.61e-2$
Example 2	0.01	$1e-5$	$2.64e-2$
	0.05	$1e-4$	$6.87e-2$
Example 3	0.01	$1e-7$	$2.29e-2$
	0.05	$1e-6$	$6.02e-2$
Example 4	0.01	$1e-3$	$3.7e-2$
	0.05	$1e-2$	$5.01e-2$

and $c(x, y) = 0$. Numerical results are listed in Table 1 and shown in Figures 3 and 4 with relative noise levels 1% and 5%, respectively.

Example 4. In this example, the exact initial value for problem (1)–(3) is taken as $\varphi(x, y) = \sin(2\pi x)\sin(2\pi y)$. And let $\alpha = 0.8$, $\theta(x, y) = 0.01(x + y)$, and $c(x, y) = -0.001$. Numerical results are listed in Table 1 and shown in Figures 5 and 6 with relative noise levels 1% and 5%, respectively.

Conflict of Interests

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

Acknowledgments

This work is supported by National Natural Science Foundation of China (11161002), Young Scientists Training Project of Jiangxi Province (no. 20122BCB23024), Natural Science Foundation of Jiangxi Province of China (no. 20142BAB201008), Ground Project of Science and Technology of Jiangxi Universities (no. KJLD14051), and National High-Tech R&D Program of China (2012AA061504).

References

[1] I. Podlubny, *Fractional Differential Equations: An Introduction to Fractional Derivatives, Fractional Differential Equations, to Methods of Their Applications*, vol. 198 of *Mathematics in Science and Engineering*, Academic Press, San Diego, Calif, USA, 1999.

[2] R. Metzler and J. Klafter, "Boundary value problems for fractional diffusion equations," *Physica A: Statistical Mechanics and its Applications*, vol. 278, no. 1-2, pp. 107–125, 2000.

[3] H. E. Roman and P. A. Alemany, "Continuous-time random walks and the fractional diffusion equation," *Journal of Physics A Mathematical and General*, vol. 27, no. 10, pp. 3407–3410, 1994.

[4] K. Sakamoto and M. Yamamoto, "Initial value/boundary value problems for fractional diffusion-wave equations and applications to some inverse problems," *Journal of Mathematical Analysis and Applications*, vol. 382, no. 1, pp. 426–447, 2011.

[5] B. Jin, R. Lazarov, and Z. Zhou, "Error estimates for a semidiscrete finite element method for fractional order parabolic equations," *SIAM Journal on Numerical Analysis*, vol. 51, no. 1, pp. 445–466, 2013.

[6] S. D. Eidelman and A. N. Kochubei, "Cauchy problem for fractional diffusion equations," *Journal of Differential Equations*, vol. 199, no. 2, pp. 211–255, 2004.

[7] Y. Lin and C. Xu, "Finite difference/spectral approximations for the time-fractional diffusion equation," *Journal of Computational Physics*, vol. 225, no. 2, pp. 1533–1552, 2007.

[8] F. Mainardi, "The fundamental solutions for the fractional diffusion-wave equation," *Applied Mathematics Letters*, vol. 9, no. 6, pp. 23–28, 1996.

[9] X. J. Li and C. J. Xu, "A space-time spectral method for the time fractional diffusion equation," *SIAM Journal on Numerical Analysis*, vol. 47, no. 3, pp. 2108–2131, 2009.

[10] O. P. Agrawal, "Solution for a fractional diffusion-wave equation defined in a bounded domain," *Nonlinear Dynamics*, vol. 29, no. 1–4, pp. 145–155, 2002.

[11] J. J. Liu and M. Yamamoto, "A backward problem for the time-fractional diffusion equation," *Applicable Analysis*, vol. 89, no. 11, pp. 1769–1788, 2010.

[12] X. T. Xiong, J. X. Wang, and M. Li, "An optimal method for fractional heat conduction problem backward in time," *Applicable Analysis*, vol. 91, no. 4, pp. 823–840, 2012.

[13] L. Y. Wang and J. J. Liu, "Data regularization for a backward time-fractional diffusion problem," *Computers & Mathematics with Applications*, vol. 64, no. 11, pp. 3613–3626, 2012.

[14] Y. Zhang and X. Xu, "Inverse source problem for a fractional diffusion equation," *Inverse Problems*, vol. 27, Article ID 035010, pp. 1–12, 2011.

[15] T. Wei and Z. Q. Zhang, "Reconstruction of a time-dependent source term in a time-fractional diffusion equation," *Engineering Analysis with Boundary Elements*, vol. 37, no. 1, pp. 23–31, 2013.

[16] J. G. Wang, Y. B. Zhou, and T. Wei, "Two regularization methods to identify a space-dependent source for the time-fractional diffusion equation," *Applied Numerical Mathematics*, vol. 68, pp. 39–57, 2013.

[17] Z.-L. Deng, X.-M. Yang, and X.L. Feng, "A mollification regularization method for a fractional-diffusion inverse heat conduction problem," *Mathematical Problems in Engineering*, vol. 2013, Article ID 109340, 9 pages, 2013.

[18] B. T. Jin and W. Rundell, "An inverse problem for a one-dimensional time-fractional diffusion problem," *Inverse Problems*, vol. 28, no. 7, Article ID 075010, pp. 1–19, 2012.

[19] Z. Wang and D. Xu, "On the linear model function method for choosing Tikhonov regularization parameters in linear ill-posed problems," *Chinese Journal of Engineering Mathematics*, vol. 30, no. 3, pp. 451–466, 2013.

[20] Z. W. Wang and J. J. Liu, "New model function methods for determining regularization parameters in linear inverse problems," *Applied Numerical Mathematics*, vol. 59, no. 10, pp. 2489–2506, 2009.

[21] Q. Chen and J. J. Liu, "Solving an inverse parabolic problem by optimization from final measurement data," *Journal of Computational and Applied Mathematics*, vol. 193, no. 1, pp. 183–203, 2006.

[22] Y. J. Jiang and J. T. Ma, "High-order finite element methods for time-fractional partial differential equations," *Journal of Computational and Applied Mathematics*, vol. 235, no. 11, pp. 3285–3290, 2011.

Hindawi Publishing Corporation
Mathematical Problems in Engineering
Volume 2014, Article ID 427203, 12 pages
http://dx.doi.org/10.1155/2014/427203

Research Article

Effective Parameter Dimension via Bayesian Model Selection in the Inverse Acoustic Scattering Problem

Abel Palafox, Marcos A. Capistrán, and J. Andrés Christen

CIMAT A.C, Jalisco S/N, Valenciana, 36240 México, GTO, Mexico

Correspondence should be addressed to Abel Palafox; abel.palafox@cimat.mx

Received 8 April 2014; Revised 23 July 2014; Accepted 23 July 2014; Published 7 September 2014

Academic Editor: Fatih Yaman

We address a prototype inverse scattering problem in the interface of applied mathematics, statistics, and scientific computing. We pose the acoustic inverse scattering problem in a Bayesian inference perspective and simulate from the posterior distribution using MCMC. The PDE forward map is implemented using high performance computing methods. We implement a standard Bayesian model selection method to estimate an effective number of Fourier coefficients that may be retrieved from noisy data within a standard formulation.

1. Introduction

Inverse problems are typically ill-posed and analytical solutions are seldom available. Approaches to inverse problems in the interface of applied mathematics, statistics, and scientific computing represent a setting with a myriad of tools for robust solutions, for a number of reasons including its sound theoretical setting for uncertainty quantification [1, 2], prediction, and decision making. See [3] for a recent review. Although these interdisciplinary approaches are becoming prevalent [4–8], topics such as knowledge representation in complex systems [9] and effective dimension are not as mature yet. Sparsity promoting regularization and Bayesian methods have received attention recently [8, 10].

In this paper we consider a nonlinear acoustic scattering inverse problem as a case study. The rationale is that the well posedness of the direct problem and robust numerical methods has been comprehensively studied [11, 12]. This fact allows us to address the inverse problem with a reliable direct problem solver.

The propagation of acoustic waves in a homogeneous isotropic medium with constant speed of sound is governed by the linear wave equation:

$$U_{tt} = c^2 \Delta U \quad \text{in } \mathbb{R}^n, n = 2, 3 \tag{1}$$

for a velocity potential U and c the speed of sound in the medium. For monochromatic time-harmonic waves with frequency ω we have

$$U(x,t) = \mathrm{Re}\left(e^{i\omega t} u(x)\right), \tag{2}$$

where the space dependent term $u(x)$ satisfies the Helmholtz equation. Consider the following:

$$\Delta u + k^2 u = 0, \tag{3}$$

where $k = \omega/c$ is called the *wave number*. Given an obstacle of compact support $D \subset \mathbb{R}^n$ $(n = 2, 3)$, its forward scattering problem is governed by the Helmholtz equation in $\mathbb{R}^n - \overline{D}$. The total wave $u(x) = \exp(ikx \cdot d) + u^s(x)$ is a superposition of the incident wave $\exp(ikx \cdot d)$ and the scattered wave $u^s(x)$, and it is subject to a boundary condition on $\Gamma = \partial D$. The boundary condition may be of type $u = 0$ (Dirichlet), $\partial u/\partial \nu = 0$ (Neumann), or $\partial u/\partial \nu + ik\lambda u = 0$ (impedance). Also, the scattered waves u^s satisfy the Sommerfeld radiation condition

$$\lim_{r \to \infty} r^{(n-1)/2} \left(\frac{\partial u^s}{\partial r} - iku^s \right) = 0, \quad r = \|x\|_2 \tag{4}$$

and are referred to as *radiating solutions* of the Helmholtz equation.

Given an incident wave $\exp(ikx\cdot d)$, the obstacle boundary Γ uniquely determines the scattered wave $u^s(x)$, and every scattered wave $u^s(x)$ which is a radiating solution of the Helmholtz equation has the asymptotic behavior of an outgoing spherical wave. Consider the following:

$$u(x) = \frac{e^{ik|x|}}{|x|}\left\{u_\infty(\widehat{x}) + O\left(\frac{1}{|x|}\right)\right\}, \tag{5}$$

where $\widehat{x} = x/|x|$ and $u_\infty(\widehat{x})$ are the scattering amplitude or *far field pattern*. We denote by F_Γ the boundary to far field mapping:

$$F_\Gamma(\widehat{x}) = u_\infty(\widehat{x}). \tag{6}$$

The mapping from the obstacle boundary Γ to the far field scattering amplitude u_∞ is injective [11, 12].

In this work we address the problem of estimating the boundary Γ given noisy measurements of the far field data u_∞. This paper is organized as follows. For the sake of making this paper self-contained, in Section 2 we briefly review useful results used throughout the paper. The acoustic layer potential and integral equation method are presented in Section 2.1. We discuss the numerical solution of the forward mapping and its implementation in Section 2.2. In Section 2.3 we present in detail the Bayesian formulation of the inverse problem along with the probability distributions involved. The inverse problem results and the effective dimension analysis are presented in Section 3. We consider the effective dimension exploration as the main contribution of this work.

2. Materials and Methods

In this section we describe the forward mapping evaluation using the integral equations method and the layer acoustic potentials approach as in [11, 12].

2.1. Single Layer and Double Layer Potentials. The fundamental solution of the Helmholtz equation (3) is

$$\Phi(x,y) = \begin{cases} \dfrac{1}{4\pi}\dfrac{e^{ik|x-y|}}{|x-y|} & n=3 \\ \dfrac{i}{4}H_0^{(1)}(k|x-y|) & n=2, \end{cases} \tag{7}$$

for $x \neq y$, where $|\cdot|$ denotes the L_2 norm and $H_0^{(1)}$ is the Hankel function of first kind and order zero. Given an integrable function φ, the integral operators

$$(S\varphi)(x) = \int_\Gamma \varphi(y)\,\Phi(x,y)\,ds(y), \quad x \in \mathbb{R}^n \setminus \Gamma, \tag{8}$$

$$(K\varphi)(x) = \int_\Gamma \varphi(y)\,\frac{\partial\Phi(x,y)}{\partial\nu(y)}\,ds(y), \quad x \in \mathbb{R}^n \setminus \Gamma, \tag{9}$$

with ν being the unit normal vector to Γ directed to the exterior of D, are called acoustic *single layer* and acoustic *double layer* potentials for density φ, respectively. Both, the single and double layer potentials are radiant solutions of the Helmholtz equation in $\mathbb{R}^n \setminus D$. The single layer potential is continuous in $\mathbb{R}^n \setminus D$ (including the boundary Γ) and consequently it is defined as in (8) also for $x \in \Gamma$. The double layer potential is only continuous in $\mathbb{R}^n \setminus \overline{D}$ (not including Γ); nonetheless $(K\varphi)(x)$ is defined in the boundary as its corresponding limit when x approaches Γ, namely,

$$(K\varphi)(x) = \frac{1}{2}\varphi(x) - \int_\Gamma \frac{\partial\Phi(x,y)}{\partial\nu(y)}\varphi(y)\,ds(y), \quad x \in \Gamma; \tag{10}$$

see [12] for details.

The direct problem is formulated in the form of a *combined potential* using the Dirichlet boundary condition:

$$\frac{1}{2}\varphi + K\varphi - i\eta S\varphi = -u^{\text{inc}}, \tag{11}$$

where $u^{\text{inc}} = \exp(ikx\cdot d)$ is the incident wave, η is a coupling factor (in our case we set $\eta = k$), and S and K are the acoustic potentials defined above.

The three potentials (single layer, double layer, and combined potential) reproduce the same far field pattern given a boundary Γ. However, the combined potential operator in (11) is the sum of the identity and a compact operator, and therefore its singular values are bounded away from zero. Consequently, the combined potential is more stable regarding numerical implementation and then we use the combined potential in the remainder.

The far field pattern for the combined potential is given by

$$u_\infty(\widehat{x}) = \frac{e^{-i(\pi/4)}}{\sqrt{8\pi k}}\int_\Gamma \{k\nu(y)\cdot\widehat{x} + \eta\}\,e^{-ik\widehat{x}\cdot y}\varphi(y)\,ds(y). \tag{12}$$

The corresponding far field pattern for the single-layer potential is given by

$$u_\infty(\widehat{x}) = \frac{e^{i(\pi/4)}}{\sqrt{8\pi k}}\int_\Gamma e^{-ik\widehat{x}\cdot y}\varphi(y)\,ds(y). \tag{13}$$

The integral (12) can be evaluated numerically using the trapezoidal rule after solving the integral equation (11) for φ. It is known [13] that the trapezoidal rule has high accuracy when integrating over periodic functions and therefore is a sensible method to use in this case.

2.2. Numerical Solution of the Forward Map. As a test case we consider the two-dimensional kite-shaped domain shown in Figure 1. The parametric representation for the boundary is given by $\Gamma(t) = (\cos t + 0.65\cos 2t - 0.65, 1.5\sin t)$, $0 \le t \le 2\pi$.

The combined potential equation (11) gives rise to a linear system upon discretization:

$$A\varphi = g, \tag{14}$$

where φ is the density, $g = -2u^{\text{inc}}$, and the matrix A has the form

$$A = I - L - ikM, \tag{15}$$

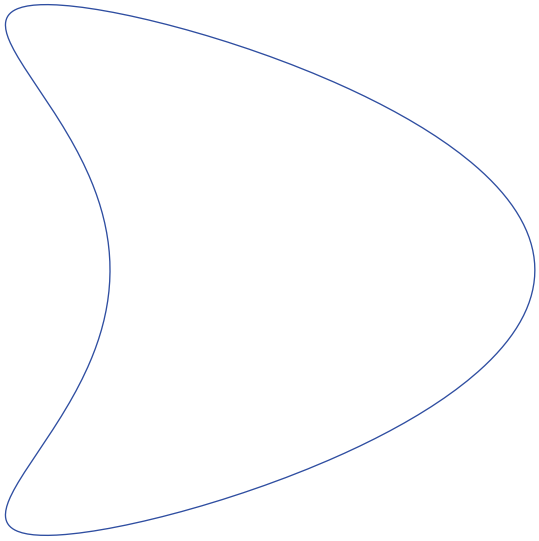


Figure 1: Synthetic example: a kite-shaped object. Numerical results for forward mapping evaluations are obtained by generating synthetic far field pattern measurements for this smooth, nonconvex boundary.

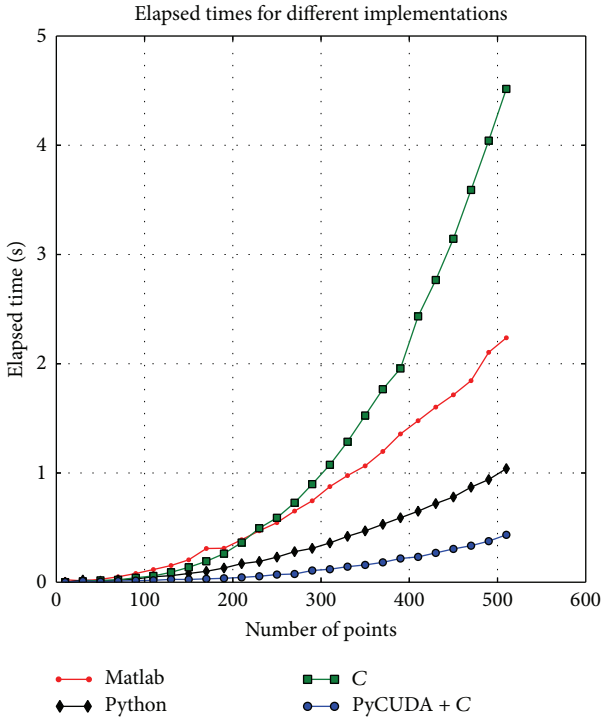


Figure 2: Elapsed times of serial and parallel numerical implementations for a single forward mapping evaluation using the combined potential.

where I is the $N \times N$ identity matrix and N is the number of knots used to discretize the boundary Γ. L and M are the discrete kernels corresponding to single layer potential and double layer potential, respectively (for details see [12]).

Matrix A in (15) is square, nonsymmetric, dense, complex, and well conditioned. Mathematical software such as Matlab or Python can be used to evaluate the direct problem.

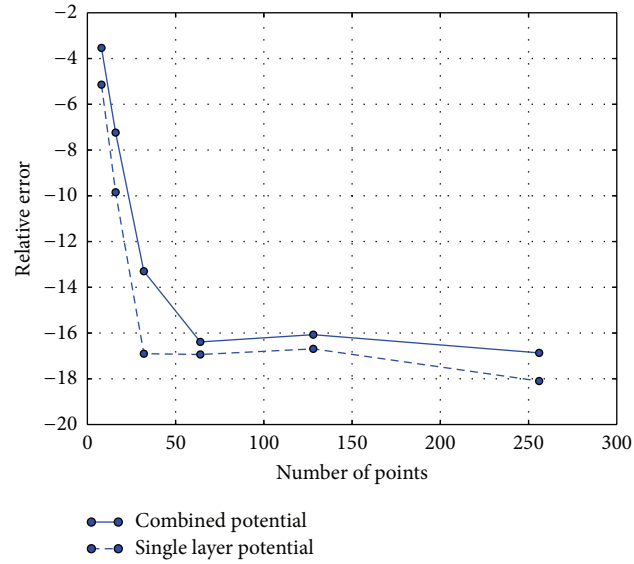


Figure 3: Relative errors of the forward mapping evaluation on direction $d = (1, 0)$, for the smooth and periodic kite-shaped boundary, varying the number of points N and wave number $k = 1$ fixed. This plot with logarithmic scale for y-axis illustrates the exponential convergence of Nÿstrom method for combined and single layer potentials.

However, it must be taken into account that a big number of evaluations of the forward mapping must be done in a Bayesian statistics approach.

Consequently, in order to have an efficient numerical machinery to evaluate the direct problem we have implemented a parallel version of the general conjugate residuals method (GCR) to solve the linear system (14). The GCR method was programmed in C on a graphics processing unit (GPU). Of note, a serial version of the GCR method programmed in C is slow, even compared with matlab and python solvers of the linear systems; see Figure 2. The implementation details for parallel computing scheme are presented on Appendix B. Although the evaluation of the far field pattern (12) is not computationally expensive, it can be done in parallel. Details are shown also in Appendix B.

The far field pattern value $u_\infty(d)$ obtained for the kite in the directions $d = (1, 0)$ closely resembles results previously reported in [12]. The convergence of Nÿstrom method, used to numerically evaluate the integral operators, is illustrated on Figure 3. We use as a reference solution the far field pattern value $u_\infty(1, 0)$ with wave number $k = 1$, obtained with the combined potential for $N = 1024$ points, and we plot in logarithmic scale for y-axis the relative errors for far field patterns obtained with combined potential (solid line in Figure 3) and single layer potential (dashed line in Figure 3) for number of points $N = 8, 16, 32, 64, 128, 256$.

2.3. Bayesian Formulation. First let us remember the Bayes rule:

$$\pi(\theta \mid u_\infty) = \frac{\pi(u_\infty \mid \theta)\,\pi(\theta)}{\pi(u_\infty)}. \tag{16}$$

The *posterior density* $\pi(\theta \mid u_\infty)$ quantifies the uncertainty of the parameter θ to be recovered from data u_∞. The *prior density* $\pi(\theta)$ expresses the information regarding the unknown parameter θ. $\pi(u_\infty \mid \theta)$ is the *likelihood* function, that is, the measurement error distribution assumed, conditional on θ. The density $\pi(u_\infty)$ is a normalizing constant. Below we describe these density functions for our inverse problem.

We consider the inverse problem defined by the nonlinear forward mapping (6) from the point of view of Bayesian statistics. In Bayesian statistical inversion theory, the solution of the inverse problem is the conditional probability of parameters, given the data (posterior probability). We assume that the data are noisy measurements of far field pattern u_∞ and that the boundary Γ may be described by a vector of parameters θ as follows.

We construct a prior model based on a parameterization for Γ considering that it is a simple 2π-periodic curve and $\Gamma \in C^2(\mathbb{R}^2)$. We express Γ as follows:

$$\Gamma(t) = r(t)(\cos t, \sin t), \tag{17}$$

where $t \in [0, 2\pi)$ and

$$r(t) = \frac{a_0}{2} + \sum_{i=1}^{\infty} a_i \cos(it) + b_i \sin(it). \tag{18}$$

Thus the radius of Γ is expressed in terms of a Fourier series. The vector of parameters

$$\theta = (a_0, a_1, \ldots, a_n, \ldots, b_1, b_2, \ldots, b_n, \ldots) \tag{19}$$

contains the Fourier series coefficients in (18). For a boundary Γ that is defined by a vector of coefficients θ we denote the forward mapping by F_θ. We refer to

$$\theta_n = (a_0, a_1, \ldots, a_n, b_1, b_2, \ldots, b_n) \tag{20}$$

as the finite vector of $c = 2n + 1$ Fourier coefficients to represent a boundary Γ_n.

2.3.1. Posterior Distribution. The posterior distribution does not have to our knowledge a closed form. Consequently, we resort to *Markov chain Monte Carlo* (MCMC) simulation methods to analyze the posterior distribution. We use the t-walk [14], a general purpose sampling algorithm for continuous distributions, in order to sample satisfactorily from the posterior distribution. We describe briefly the t-walk algorithm and implementation details in Appendix A.

The t-walk algorithm evaluates the energy function (minus log of the nonnormalized posterior). Consider the following:

$$\varepsilon(\theta) = E - \log L(\theta, u_\infty) - \log \pi_\Theta(\theta), \tag{21}$$

for an arbitrary constant E (i.e., minus log of the nonnormalized posterior) on each iteration. In practice the constant E is not used and the energy is evaluated as minus log of the product likelihood times the prior one.

The output simulations of the t-walk are vectors of Fourier coefficients θ_n for n fixed, each one of which defines a boundary Γ_n through (18). We refer subsequently to the simulations θ_n, obtained with the t-walk, as MCMC simulations or posterior samples and we refer to a *simulated boundary* as the curve Γ_n which is defined by a simulated vector θ_n.

2.3.2. Prior Distribution. We establish the prior model as follows. Suppose that $\Gamma \in C^2(\mathbb{R}^2)$ corresponds to a star shaped domain. Then Γ admits a representation (17)-(18). Furthermore, the mean square error for approximating Γ by $\widetilde{\Gamma}_n$ is given by

$$\left\|\Gamma - \widetilde{\Gamma}_n\right\|^2 = \sum_{m>n} \left|(a_m, b_m)\right|^2. \tag{22}$$

Further details of this consequence of Parseval inequality are described in [15].

Due to smoothness of Γ, the amplitude of the Fourier coefficients decays quickly along the series. In fact

$$\left|(a_m, b_m)\right| = O\left(m^{-2}\right). \tag{23}$$

Note that m is the position of the coefficient in the Fourier series. Therefore the norm $|(a_m, b_m)|$ of the coefficients a_m, b_m is by definition of $O(m^{-2})$ bounded by

$$\left|(a_m, b_m)\right| < \frac{C}{m^2}, \tag{24}$$

for some positive constant $C \in \mathbb{R}$.

We pose a normal distribution for each pair a_m, b_m of coefficients as follows:

$$a_m, b_m \sim N\left(0, \frac{C}{4m^2}\right), \tag{25}$$

with a_m, b_m mutually independent. We note that the variance was set to $C/4m^2$. The rationale is that with the variance in this way, we guarantee that the pair a_m, b_m satisfies the inequality (24) with probability higher than 0.99. Moreover the variance decreases when increasing position m, that is, for high order coefficients along the Fourier series. Consequently, this modeling incorporates the smoothness of Γ on the prior distribution.

The coefficient a_0 in (18) is related to the size of the scattering object. We assume that

$$a_0 \sim N(2.5, 0.14). \tag{26}$$

It is not clear how to set the constant C in distribution (25). This constant can be seen as a scaling factor and the parameter m controls the spread of the distribution by $1\backslash 4m^2$. A single constant C can be chosen for all prior distributions (25) independently of the value of the index $m = 1, \ldots, n$. For this aim we sample coefficients $\widetilde{\theta}_n$ from the distributions (25) varying the value of C. We add the condition

$$r(t) > 0.75 \quad \forall t \in [0, 2\pi) \tag{27}$$

on the radius (18) to guarantee that the sample curves do not intersect themselves. The sampled curves are presented on Figure 4. As we observe, using the values of $C = 0.5$ and $C = 1$ (see Figures 4(a) and 4(b)) we obtain very smooth curves. On the other hand, Figures 4(c) and 4(d) show curves with more oscillations. For our purposes, a value of $C = 2$ is chosen.

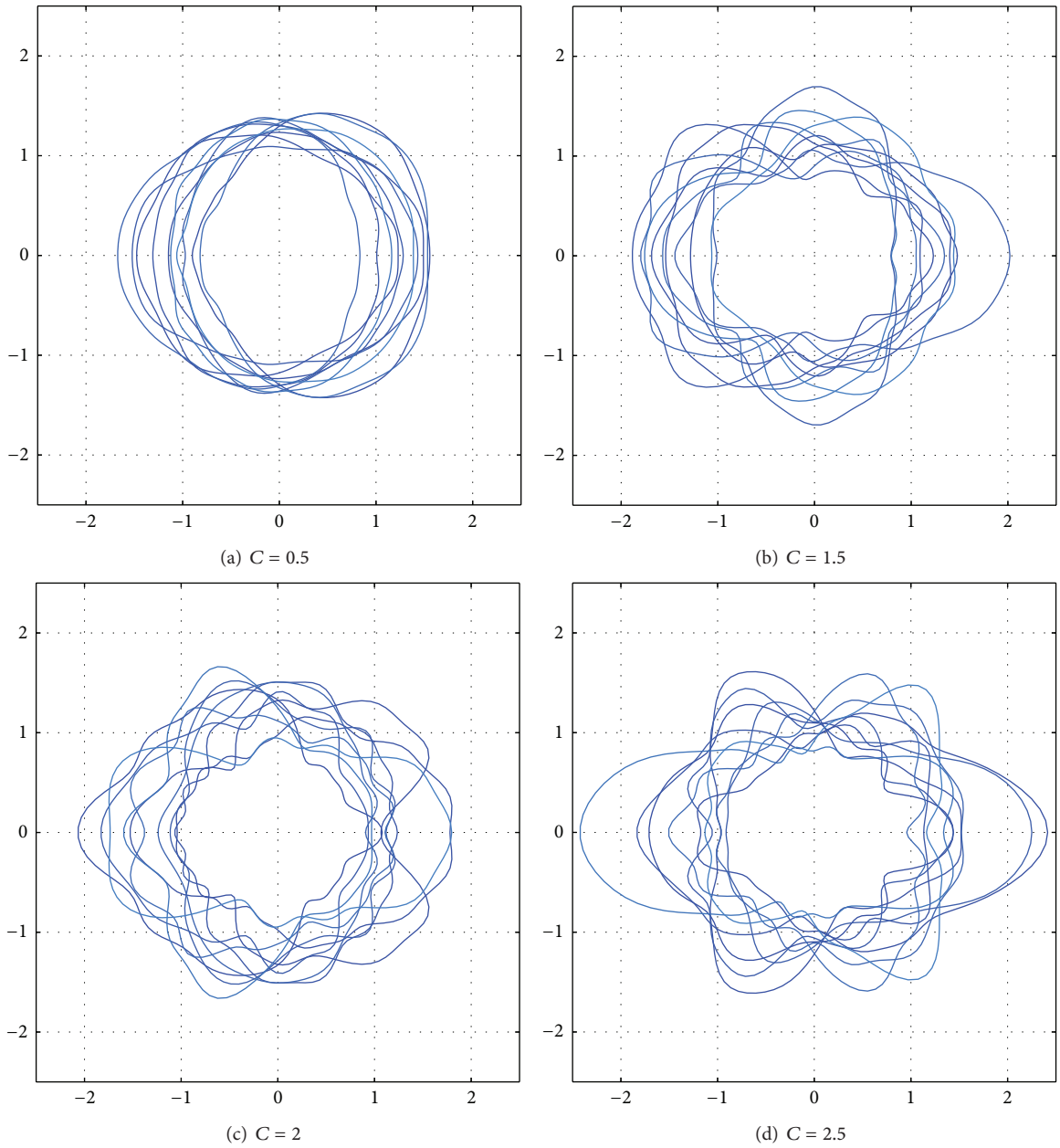


FIGURE 4: Prior distribution samples for coefficients on our boundary representation. The sample is drawn from a Gaussian distribution with mean $\mu = 0$ and variance $\sigma^2 = C/4m^2$, where m is the position of the coefficient on a Fourier series and C is constant. We vary the variance and set the value of C depending on the expected smoothness of the boundary to recover.

2.3.3. Likelihood. Assume that the measurements of the far field pattern have additive Gaussian noise

$$u_\infty(\hat{x}) = F^\Gamma(\theta) + \eta_{\hat{x}}, \tag{28}$$

with $\eta_{\hat{x}} \sim N(0, \sigma^2)$ for a boundary Γ. The likelihood is given by the noise distribution

$$L(\theta, u_\infty) = (2\pi\sigma)^{-N/2} \exp\left\{-\frac{1}{2}\sum_{\hat{x}}\left\|\frac{u_\infty(\hat{x}) - F^\Gamma(\hat{x})}{\sigma}\right\|^2\right\}, \tag{29}$$

assuming independent measurements. We assume that the measurements u_∞ are taken on evenly spaced directions $\hat{x}$.

3. Results and Discussion

In order to avoid committing an "inverse crime" [5] in our numerical experiments we proceed as follows. First we generate synthetic noisy far field measurements from the kite-shaped object (see Figure 1). Second, this kite is not obtained from any finite Fourier series expansion and it is therefore not included in any of our models. This forces our

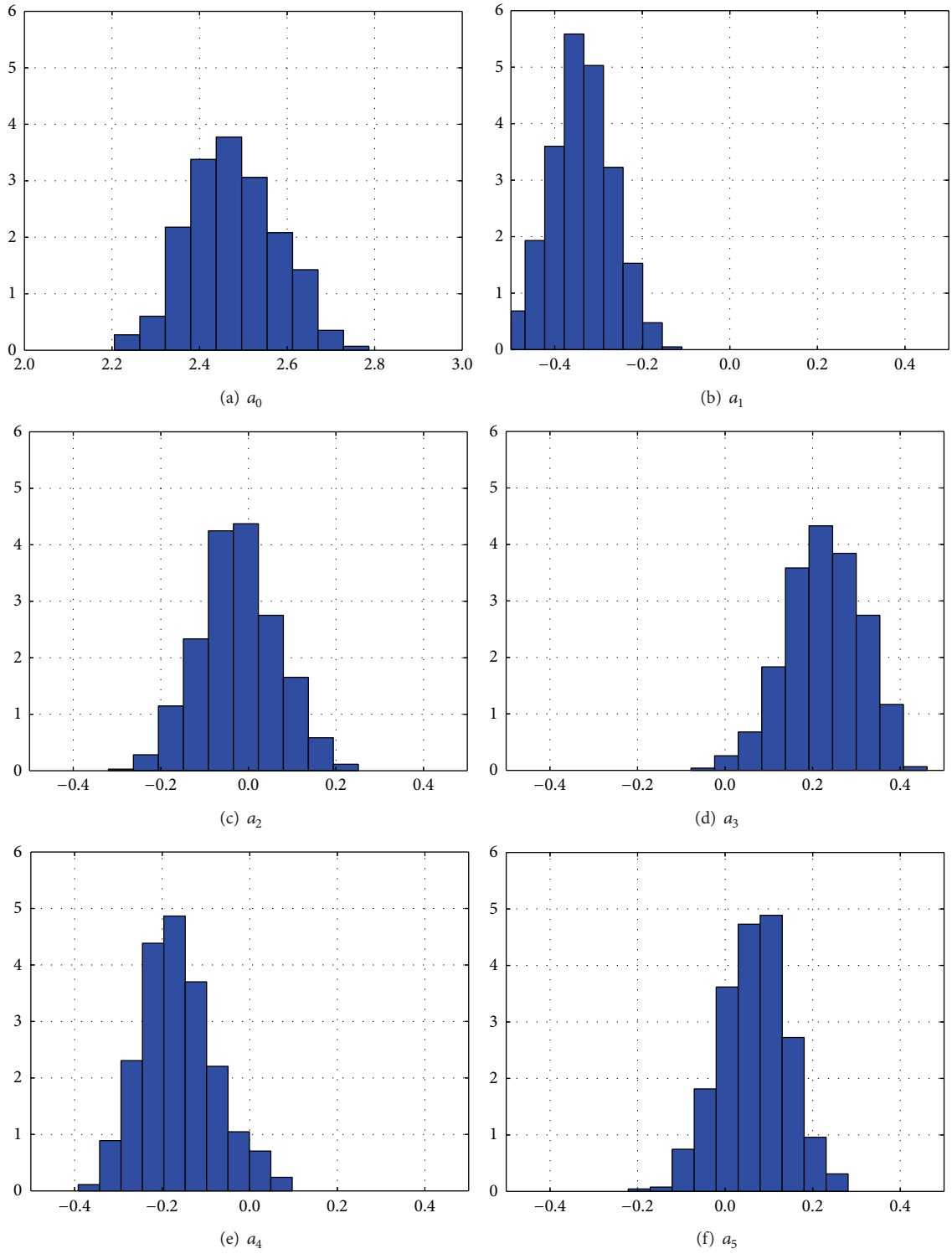
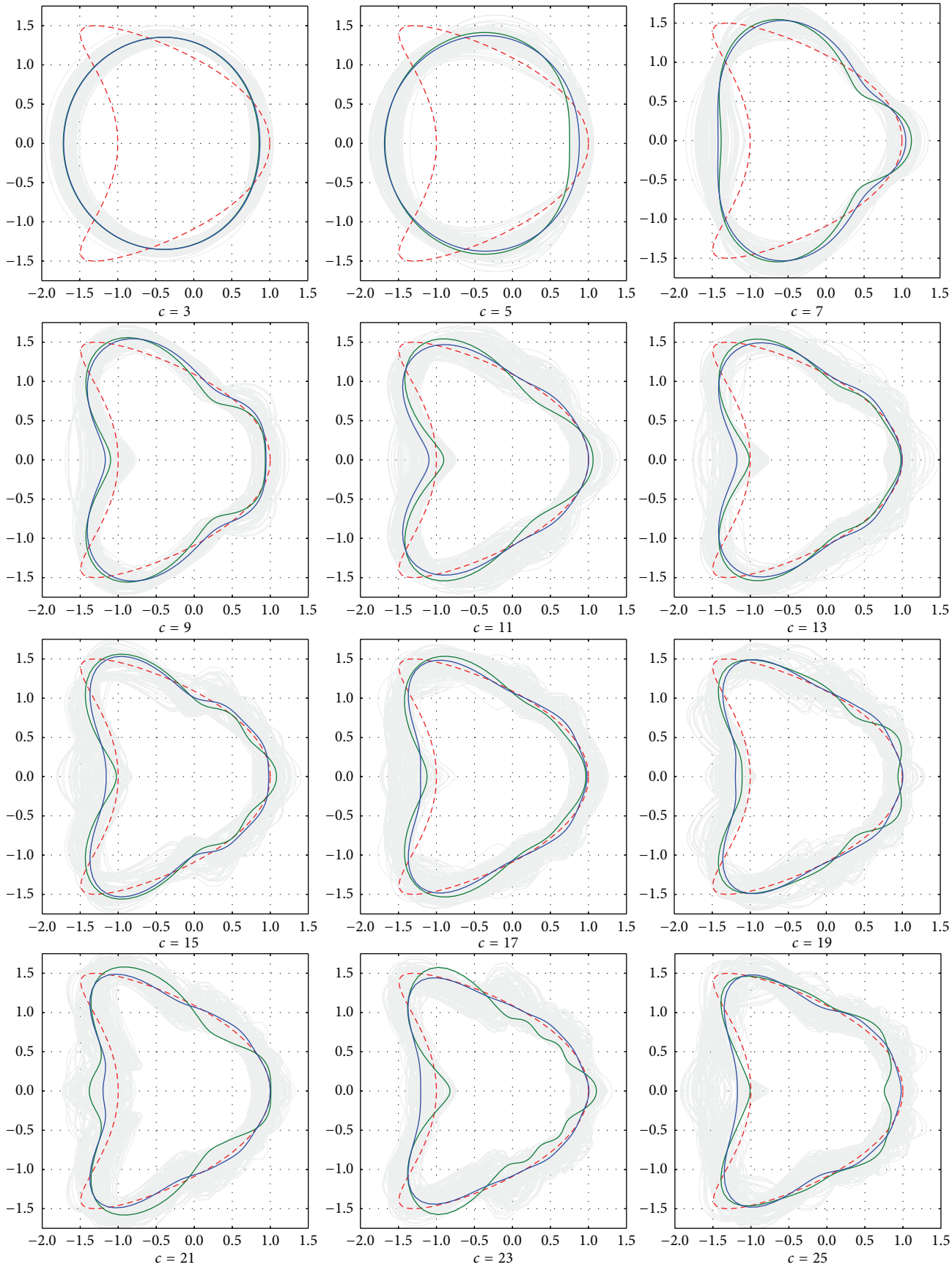


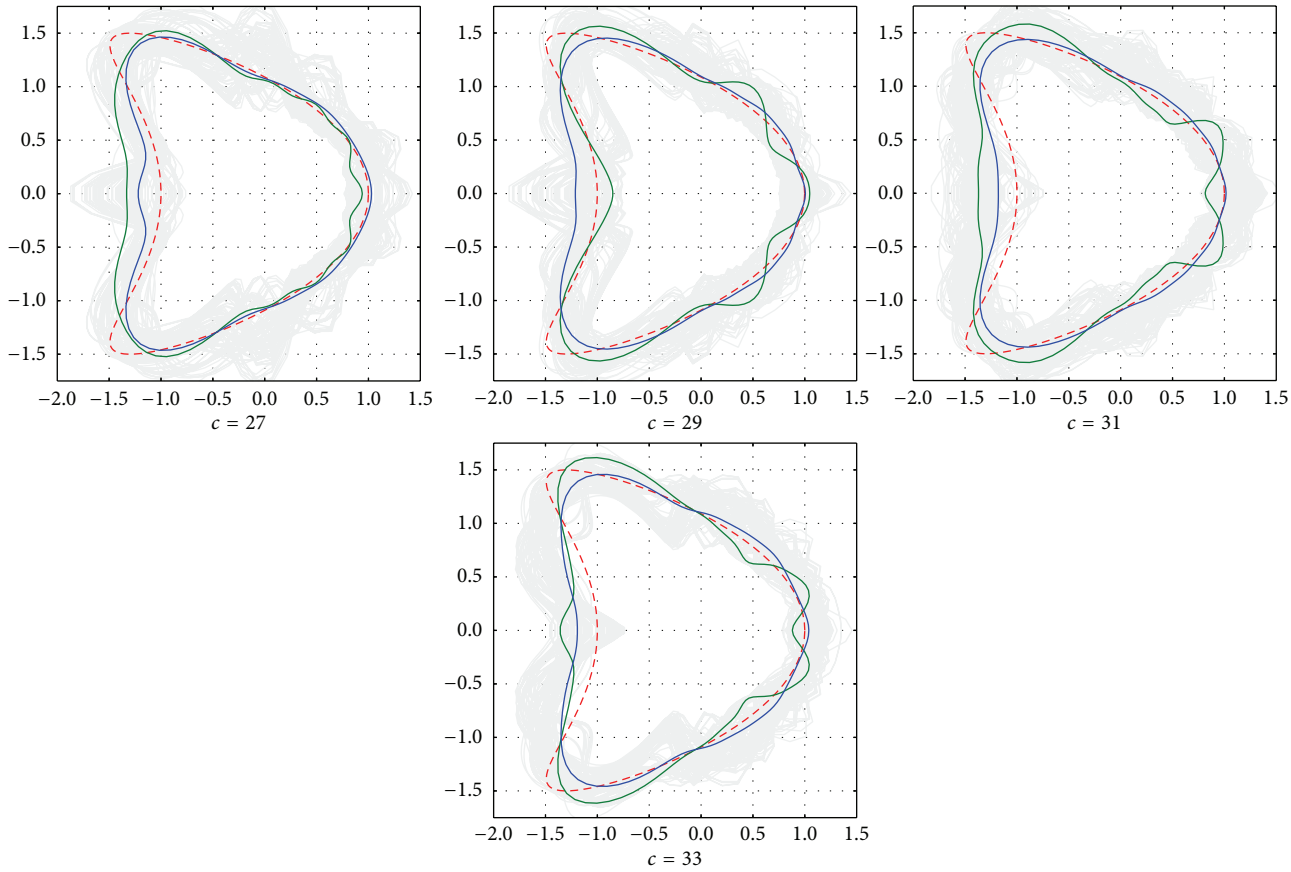
FIGURE 5: Marginal posterior distributions for a_i's coefficients for $c = 11$. These unimodal distributions are used to estimate coefficients of the inverse problem.

methodology to cope with diverse shapes, being the finite Fourier series only an approximation. Third, synthetic data generation is done using the single-layer potential approach, whereas for MCMC steps we evaluate the forward mapping using the combined potential. Of note, both the single and the combined potential approaches approximate the same far field pattern when N is large enough, however they are numerically different.



(a)

Figure 6: Continued.



(b)

Figure 6: Probability Regions from the MCMC simulations (gray). The boundaries defined from mean of marginal posterior distribution are shown in blue. The corresponding MAP boundaries are shown in green. The results are compared to the kite object in dashed red line.

We present results of the MCMC simulations, varying the number of Fourier coefficients in Section 3.1. Results representing the main contribution of this paper are presented on Section 3.2.

3.1. Inverse Problem Results. For our experiments we discretize the boundary of the kite-shaped object with 256 points. The far field pattern was computed on 16 evenly spaced points. We use 16 incident waves and we set the coupling parameter η to be equal to the wave number $k = 1$. The synthetic data are generated with additive Gaussian noise $N(0, \sigma^2)$ whit $\sigma = 0.1$ (equivalently SNR = 0.97). We generate 100,000 MCMC simulations of the posterior distribution for each number of coefficients ($c = 3, 5, \ldots, 31, 33$). The transient stage of the MCMC is taken into account and we remove the first 5,000 iterations in each case. We highlight that it takes about 7 hours to produce 100,000 MCMC simulations on a Python implementation. With our parallel implementation the execution time is reduced to about 1 hour.

Due to space constrains, we present only marginal posterior probability distributions of cosine terms for $c = 11$ coefficients in Figure 5. As we observe, for each coefficient, the corresponding distribution is unimodal with low variation. The same feature is presented for coefficients for cases $c \in \{3, 5, \ldots, 31, 33\}$, $c \neq 11$ (not shown).

We present in Figure 6 the probability region (in gray) for the number of Fourier coefficients for cases $c \in \{3, 5, \ldots, 31, 33\}$. For this aim we draw 1000 simulated boundaries subsampled with their corresponding IAT factor in each case. The true kite-shaped boundary is shown as a red dashed line. From Figure 6 it is apparent that as we increase the number of coefficients the samples have more oscillations and the probability region becomes imprecise. This fact exhibits the numerical instability of Fourier-based representation for the solutions of the inverse problem of interest.

Also, in Figure 6 we show what we refer to as MAP *boundary* (dashed line) and *mean boundary* (continuous line) for each value of c. The MAP boundary is the curve defined by the simulated vector θ_n that has the lowest energy value (maximum a posteriori). The mean boundary is the curve defined by the mean of each coefficient of simulated vectors θ_n. MAP boundaries and mean boundaries seem to approximate the true curve with approximately the same quality. In fact, we cannot qualitatively distinguish the best approximation from $c = 9$ coefficients to $c = 33$. This issue gives rise to the analysis presented on the next section.

3.2. Effective Dimension. We refer to *effective dimension* as the number of parameters that can be properly retrieved from a noisy data set. In the context of this paper, effective dimension

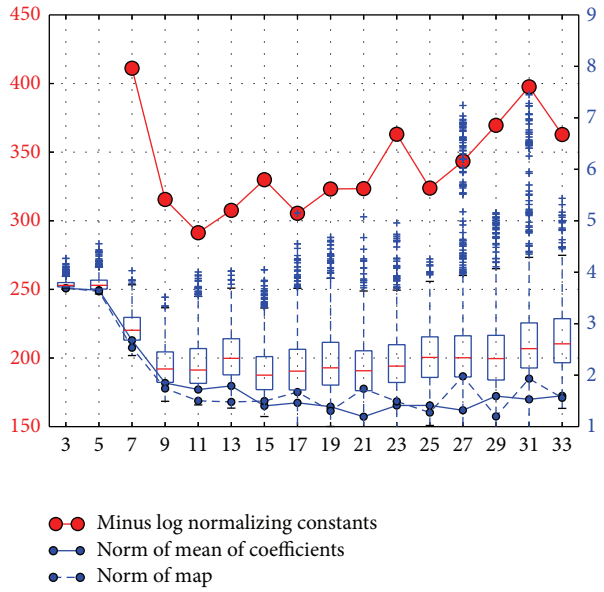


FIGURE 7: Minus logarithm of estimated normalizing constants for Bayesian model comparisons. Best model refers to the highest normalizing constant ($c = 11$ for this case). Continuous line shows the distance between the kite and the mean boundaries. Dashed line corresponds to the distance between the kite and the MAP. The boxplot corresponds to distances of the last 1000 simulations of the posterior to the kite. Red scale corresponds to minus logarithm of normalizing constants and blue scale is for the distances.

is related to the uncertainty quantification of parameters, that is, identifying the number of Fourier coefficients on the representation given by (17)–(20), required to approximate properly the boundary of the kite (see Figure 1) from synthetic far field measurements. As an aside comment we want to mention that effective dimension can be seen as an indicator for a parsimonious approximation regarding computational cost. However this feature is not used here due to the fact that effective dimension analysis is performed as a posterior procedure. In this section we have changed the notation for the distributions to be consistent with [14].

We define a collection of k models given by

$$\pi^n_{u^\infty|\Theta_n}\left(u^\infty \mid \theta_n\right), \tag{30}$$

with $n \in \{1, 2, \ldots, k\}$. Θ_n is a vector of random variables. A realization of Θ_n is a vector θ_n defined in (20), and $\pi^n_{u^\infty|\Theta_n}$ is the corresponding posterior distribution for model n. For simplicity we use as index n to indicate the model corresponding to $c = 2n + 1$ Fourier coefficients.

We use the super-model approach; that is, a new model is defined

$$\pi_{u^\infty|\Theta,M}\left(u^\infty \mid \theta, n\right) = \pi^n_{u^\infty|\Theta_n}\left(u^\infty \mid \theta_n\right), \tag{31}$$

where $M = n$ is the indicator of the model and $\Theta = (\Theta_1, \Theta_2, \ldots, \Theta_k)$.

The posterior probability for model $M = n$ of explaining the observed far field pattern is given by

$$\pi\left(M = n \mid u_\infty\right) \propto \pi_M(n)\, \pi^n_{u_\infty}\left(u_\infty\right), \tag{32}$$

where $\pi_M(n)$ is the prior probability of model $M = n$ and $\pi^n_{u_\infty}(u_\infty)$ is the corresponding normalizing constant (also known as the marginal likelihood) for model n. A straightforward way to perform model selection is by computing (32) for all the models and choosing the model with highest probability. Indeed, the difficulty regarding the computation of (32) is evaluating the normalizing constant $\pi^n_{u_\infty}(u_\infty)$. To perform our MCMC only the nonnormalized posterior is needed and therefore $\pi^n_{u_\infty}(u_\infty)$ was not previously calculated (this is the usual case when doing MCMC in a fixed number of parameters in a Bayesian application). Nevertheless, the MCMC simulations may be used to estimate $\pi^n_{u_\infty}(u_\infty)$ and provide information to explore the effective dimension. This is explained below.

The ratio of two normalizing constants is defined as follows:

$$r = \frac{\pi^l_{u^\infty}\left(u^\infty\right)}{\pi^s_{u^\infty}\left(u^\infty\right)}, \tag{33}$$

where $s, l \in \{1, 2, \ldots, k\}$, $s \neq l$. Considering a uniform prior distribution for the models, that is, $\pi_M(n) = 1 \setminus n$, model s is better than model l when $r > 1$, and they are indistinguishable if $r = 1$ and l is better than s when $r < 1$. Then, the best model has the highest normalizing constant value.

A consistent estimator of the ratio is

$$\widehat{r} = \frac{(1/m_l)\sum_{i=1}^{m_l} \pi^l_{u^\infty|\Theta_l}\left(u^\infty \mid \theta_{l,i}\right) \pi^l_{\Theta_l}\left(\theta_{l,i}\right) / \pi^I_l\left(\theta_{l,i}\right)}{(1/m_s)\sum_{i=1}^{m_s} \pi^s_{u^\infty|\Theta_s}\left(u^\infty \mid \theta_{s,i}\right) \pi^s_{\Theta_s}\left(\theta_{s,i}\right) / \pi^I_s\left(\theta_{s,i}\right)}, \tag{34}$$

where π^I_l, π^I_s are completely known importance sampling densities for $\pi^l_{\Theta_l|u^\infty}$ and $\pi^s_{\Theta_s|u^\infty}$, respectively, and the sets $\{\theta_{l,1}, \theta_{l,2}, \ldots, \theta_{l,m_l}\}$ and $\{\theta_{s,1}, \theta_{s,2}, \ldots, \theta_{s,m_s}\}$ are samples of the posterior distribution for models l and s, respectively, (see [14] for details).

We observe from (34) that for a model s, the product

$$\begin{aligned} &\pi^s_{u^\infty|\Theta_s}\left(u^\infty \mid \theta_{s,i}\right) \pi^s_{\Theta_s}\left(\theta_{s,i}\right) \\ &\text{for each } \theta_{s,i} \in \left\{\theta_{s,1}, \theta_{s,2}, \ldots, \theta_{s,m_s}\right\}, \end{aligned} \tag{35}$$

is the corresponding likelihood times prior distribution. This product is in fact the energy (21) using a suitable constant E. Then the resulting sample from MCMC posterior simulations is used to compute (34), as a byproduct of the former with marginal computational cost added. Some details about approximating normalizing constant from MCMC simulations are discussed in [16].

The importance sampling density π^I_s must be comparable to the product (35). We propose to numerically estimate the density π^I_s for every model using a Kernel density estimation (KDE) [17] of the posterior distribution using the MCMC sample. A rough estimate may be used, for the sole purpose of estimating r.

The kernel density estimator has the form

$$\widehat{\pi}_s\left(\theta^*_s \mid u_\infty\right) = \frac{1}{m_s h_{m_s}} \sum_{i=1}^{m_s} K_s\left(\frac{\theta^*_s - \theta_{s,i}}{h^i_{m_s}}\right), \tag{36}$$

where the kernel K is a bounded density on R^s and h_{m_s} is the bandwidth. For this estimator we define a Gaussian kernel

$$K_s(x) = (2\pi)^{-s/2} e^{-x^2/2} \tag{37}$$

and a bandwidth

$$h_{m_s} = \prod_{i=1}^{m_s} h^i_{m_s}, \tag{38}$$

where $h^i_{m_s} = 1.06\sigma_i m_s^{-1/5}$ and σ_i is the standard deviation for each coefficient on the model s. We compute

$$c_s = \left(\frac{1}{m_s}\right) \sum_{i=1}^{m_s} \frac{\pi^s_{u^\infty|\Theta_s}\left(u^\infty \mid \theta_{s,i}\right) \pi^s_{\Theta_s}\left(\theta_{s,i}\right)}{\pi^I_s\left(\theta_{s,i}\right)}, \tag{39}$$

for each model $s \in \{1, 2, \ldots, k\}$ then all the models can be compared together. Then effective dimension corresponds to the model with the highest normalizing constant, equivalently the lowest value of minus logarithm of the normalizing constant.

We present in Figure 7 the minus logarithm of the estimated normalizing constants (39) for all models (red line). As we see in this figure, the lowest value is obtained with $n = 11$ coefficients. Also we present in the same Figure the distance (norm L_2 of the pointwise difference) between the radius of the kite and the radius of the simulated boundaries. The continuous line shows the difference with respect to the mean boundaries. The dashed line corresponds to the distances with respect to the map boundaries. The boxplot in the same figure corresponds to distances of the last 1000 simulations of the posterior. These last simulations are taken and subsampled with their corresponding IAT factor. These boundaries are in fact the probability region shown in Figure 6.

Based on the Bayes factors, the best model corresponds to $n = 11$ (we recall that $c = 2n + 1$). Of note, there are MAP boundaries (e.g. $c \in \{13, 19, 25, 29\}$) with lower mean squared error compared with the MAP for $c = 11$. Also the mean boundaries from $c = 15$ to $c = 33$ have lower distances to the MAP. However as we observe on the boxplot, the mean of the error decreases from 3 to 11 Fourier coefficients and then oscillations appear, increasing spread and increasing error from 13 to 33 coefficients. Despite the low error of MAP and mean boundaries for models with 13 to 33 coefficients, due to the spread shown in the boxplot, the best model is certainly given by Bayes' factor (i.e., for $c = 11$). A qualitative analysis that agrees with these results is also discussed in Section 3.

Fourier series is a classical representation for smooth periodic closed curves. Furthermore, we have used a well understood and high order numerical method for forward mapping evaluation. However, these two elements by themselves are not enough to have confidence regarding the solution of the inverse problem. The quality of the solution depends also on the number of coefficients of the Fourier series used for the representation.

Our effective dimension discussion relies on MCMC methods to provide a quantitative strategy to determine the number of parameters that can be retrieved given a data set.

4. Conclusions

In this work we address the acoustic inverse scattering problem with a classical Fourier-based representation of the solution. We pose the inverse problem as a Bayesian inference problem and use the output of a MCMC method (namely, the t-walk) for our effective dimension results. For the corresponding direct problem we have used the classical layer potential approach, which was solved in a fast and reliable manner with a robust numerical method and parallel computing. Using Fourier series to represent solutions allows for a straightforward formulation that incorporates the smoothness of the solutions into the prior distribution. On the other hand, the finite Fourier representation is numerically unstable. Although other approaches to represent the scattering obstacle are applicable (e.g., wavelet basis which correspond to Besov priors), a fundamental question remains: How much information can be retrieved, within the representation, from a noisy data set?

The main contribution of this paper is the effective dimension method, which is a quantitative method to estimate and quantify the uncertainty of the estimable parameters given a noisy data set. Given a parametric representation of the solution of the inverse problem, the effective dimension method is implemented via Bayesian model selection where the normalizing constant for each model is approximated using the MCMC output. Of note, the effective dimension method is applicable regardless of the parametric representation of the solution.

Appendices

A. The t-Walk

The t-walk (for "traverse" or "thoughtful" walk) is a MCMC sampler for arbitrary continuous distributions that require no tuning. The t-walk maintains two independent points in the sample space and all moves are based on four proposals (walk, traverse, hop, and blow) that are accepted with a standard Metropolis-Hastings acceptance probability on the product space. These moves produce an efficient sampling algorithm that is invariant to scale and approximately invariant to affine transformations of the state space.

For an objective function (e.g., posterior distribution) $\pi(x)$, $x \in \mathcal{X}$, and $\mathcal{X} \subset \mathbb{R}^n$, a new objective function is defined as $f(x, x') = \pi(x)\pi(x')$ in the corresponding product space $\mathcal{X} \times \mathcal{X}$. Then two proposals are considered as follows:

$$\left(y, y'\right) = \begin{cases} \left(x, h\left(x, x'\right)\right), & \text{with probability 0.5} \\ \left(h\left(x, x'\right), x'\right), & \text{with probability 0.5,} \end{cases} \tag{A.1}$$

where $h(\cdot, \cdot)$ is defined by one of the following four moves.

(i) Walk Move. The walk move is defined by the function

$$h_w\left(x, x'\right)_j = \begin{cases} x_j + \left(x_j - x'_j\right)\alpha_j, & I_j = 1 \\ x_j, & I_j = 0, \end{cases} \tag{A.2}$$

for $j = 1, 2, \ldots, n$, where $\alpha_j \in \mathbb{R}$ are i.i.d. r.v. with density

$$\psi_w\left(\alpha_j\right) = \begin{cases} \dfrac{1}{k\sqrt{1+\alpha_j}}, & \alpha \in \left[\dfrac{-a_w}{1+a_w}, a_w\right] \\ 0, & \text{otherwise} \end{cases} \tag{A.3}$$

with $a_w = 1.5$, $k = 2(\sqrt{1+a_w} - 1/\sqrt{1+a_w})$, and $\alpha_j = (a_w/(1+a_w))(-1+2u+a_w u^2)$ with $u \sim U(0,1)$. In this case the Hastings ratio is

$$\frac{g\left(x \mid y, x'\right)}{g\left(y \mid x, x'\right)} = 1, \tag{A.4}$$

for both cases on (A.1).

(ii) Traverse Move. The traverse move is defined by the function

$$h_t\left(x, x'\right)_j = \begin{cases} x'_j + \beta\left(x'_j - x_j\right), & I_j = 1 \\ x_j & I_j = 0, \end{cases} \tag{A.5}$$

where $\beta \in \mathbb{R}^+$ is a r.v. with density

$$\begin{aligned} \psi_t\left(\beta\right) = {} & \frac{a_t - 1}{2a_t}\left\{\left(a_t + 1\right)\beta^{a_t} I_{(0,1]}\left(\beta\right)\right\} \\ & + \frac{a_t + 1}{2a_t}\left\{\left(a_t - 1\right)\beta^{-a_t} I_{(1,\infty]}\left(\beta\right)\right\} \end{aligned} \tag{A.6}$$

which is a mixture of densities and can be easily sampled as follows:

$$\beta = \begin{cases} \dfrac{1}{u^{a_t+1}}, & \text{with probability } \dfrac{a_t - 1}{2a_t} \\ \dfrac{1}{u^{1-a_t}}, & \text{with probability } \dfrac{a_t + 1}{2a_t}, \end{cases} \tag{A.7}$$

with $a_t = 6$ and $u \sim U(0,1)$. The acceptance ratios are

$$\frac{\pi\left(y'\right)}{\pi\left(x'\right)}\beta^{n_I - 2} \quad \text{or} \quad \frac{\pi\left(y\right)}{\pi\left(x\right)}\beta^{n_I - 2}, \tag{A.8}$$

for first and second cases on (A.1), respectively.

(iii) Hop Move. The hop move is defined by the function

$$h_h\left(x, x'\right) = \begin{cases} x_j + \dfrac{\sigma\left(x, x'\right)}{3} z_j, & I_j = 1 \\ x_j, & I_j = 0, \end{cases} \tag{A.9}$$

with $z_j \sim N(0,1)$ and $\sigma(x, x') = \max_{I_j=1}|x_j - x'_j|$. For this proposal

$$\begin{aligned} & g_h\left(y \mid x, x'\right) \\ & = \frac{(2\pi)^{-n_I/2} 3^{n_I}}{\sigma(x, x')^{n_I}} \exp\left\{-\frac{9}{2\sigma(x, x')^2}\sum_{I_j=1}\left(y_j - x_j\right)^2\right\} \\ & \quad \times \prod_{I_j=0}\delta_{x_j}\left(y_j\right). \end{aligned} \tag{A.10}$$

(iv) Blow Move. The blow move is defined by the function

$$h_h\left(x, x'\right) = \begin{cases} x'_j + \sigma\left(x, x'\right) z_j, & I_j = 1 \\ x_j, & I_j = 0, \end{cases} \tag{A.11}$$

with $z_j \sim N(0,1)$ and $\sigma(x, x') = \max_{I_j=1}|x_j - x'_j|$. For this proposal

$$\begin{aligned} & g_b\left(y \mid x, x'\right) \\ & = \frac{(2\pi)^{-n_I/2}}{\sigma(x, x')^{n_I}} \exp\left\{-\frac{1}{2\sigma(x, x')^2}\sum_{I_j=1}\left(y_j - x_j\right)^2\right\} \\ & \quad \times \prod_{I_j=0}\delta_{x_j}\left(y_j\right). \end{aligned} \tag{A.12}$$

The numerical implementation of the algorithm only requires the user to define three functions.

(i) *Initialization*. A function to generate the two different initial guesses x and x'.

(ii) *Support*. Defines the support of the target function. Points outside of the support are rejected.

(iii) *Energy*. In this function the minus logarithm of the target density is evaluated.

The algorithm is available for download from Andres Christen's personal web page http://www.cimat.mx/-jac/twalk and it has been implemented on C, C++, Matlab, R, and Python languages.

B. Parallel Computing

Although MCMC methods are by definition serial procedures, the high computational cost can be reduced by performing each evaluation of the objective function in a parallel computing scheme when possible. In this appendix we present a way to solve the direct problem for combined potential and Nÿstrom method.

We recall the linear system matrix for combined potential and Nÿstrom method:

$$\left(I - L + ikM\right)\varphi = g, \tag{B.1}$$

where L, M are kernels for double layer potential and single layer potential (for details see [12]). The matrix coefficient ij corresponds to

$$\left(I - L + ikM\right)_{ij} = \left(I_{ij} - L_{ij} + ikM_{ij}\right), \tag{B.2}$$

where $I_{ij} = 0$ for $i \neq j$ and $I_{ij} = 1$ for $i = j$, $L_{ij} = L(t_i, t_j)$, and $M_{ij} = M(t_i, t_j)$, with $t_i = 2\pi i/N$, $t_j = 2\pi j/N$, and $i, j = 0, 1, \ldots, N$.

The evaluation of each entry of the matrix involves the numerical evaluation of Bessel functions which is computationally costly. On the other hand, each entry of the matrix is independent. Then the matrix setup is recommended to be performed in a parallel scheme in order to reduce the

$r_0 = -u_i - A\varphi_0$
$p_0 = r_0$
for $i < N$ **do**
$$a_i = \frac{(r_i, Ap_i)}{(Ap_i, Ap_i)}$$
$x_{i+1} = x_i + a_i p_i$
$r_{i+1} = r_i - a_i A p_i$
$$p_{i+1} = r_i + \sum_{j=0}^{i} b_j^{(i)} p_j, \text{ where } b_j^{(i)} = -\frac{\left(Ar_{i+1}, Ap_j\right)}{\left(Ap_j, Ap_j\right)}$$
end for

Algorithm 1: Conjugate residuals method (GCR).

computing time. That is, given a number of knots N we define a grid of threads in CUDA of size $N \times N$ and each entry ij is evaluated in a different thread.

In order to solve the linear system (B.1) we choose the GCR method which is described in Algorithm 1. The dot products involved are performed based on the cascading algorithm of the *optimizing parallel reduction in* CUDA sample from Nvidia CUDA toolkit documentation (see web page http://docs.nvidia.com/cuda).

The most demanding part is computing the vector p since a set of i coefficients b should be computed on each iteration. For this aim, we use i CUDA blocks and we compute the corresponding $b_j^{(i)}$ within each block by using cascading algorithm.

When using multiple incident waves, the term u_i becomes a matrix. This is equivalent to solve as many linear systems as incident waves. The same algorithm is performed using a CUDA block for each incident wave excepting the computing of $b_j^{(i)}$ which is done for a single incident wave at time. The computing of the far field pattern can be done in parallel by approximating the integral (12) for the multiple incident waves. However this latter evaluation is not expensive and it takes no advantage of the parallel computing.

In our experiments, this parallel computing implementation has a performance enhancement of about 7x speedup over the original. An implementation with a more effective method than this parallel version of GCR is left as future work.

Conflict of Interests

The authors declare that they have no conflict of interests regarding the publication of this paper.

Acknowledgments

The authors would like to thank Editor Fatih Yaman and anonymous referees for their constructive and detailed remarks which have helped to improve this paper in a major way. This research was supported by Grant GTO-2011-C04-168676 Guanajuato-CONACyT.

References

[1] T. Oden, R. Moser, and O. Ghattas, "Computer predictions with quantified uncertainty, part I," *SIAM News*, vol. 43, no. 9, 2010.

[2] T. Oden, R. Moser, and O. Ghattas, "Computer predictions with quantified uncertainty, part II," *SIAM News*, vol. 43, no. 10, pp. 1–4, 2010.

[3] C. Fox, H. Haario, and J. Christen, *Bayesian Theory and Applications*, Inverse Problems, 2013.

[4] S. Geman and D. Geman, "Stochastic relaxation, Gibbs distributions and the Bayesian restoration of images," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 6, no. 6, pp. 721–741, 1984.

[5] J. Kaipio and E. Somersalo, *Statistical and Computational Inverse Problems*, vol. 160 of *Applied Mathematical Sciences*, Springer, New York, NY, USA, 2005.

[6] A. Tarantola, *Inverse Problem Theory and Methods for Model Parameter Estimation*, SIAM, Philadelphia, Pa, USA, 2005.

[7] L. Biegler, G. Biros, O. Ghattas et al., Eds., *Large-Scale Inverse Problems and Quantification of Uncertainty*, Wiley Series in Computational Statistics, John Wiley & Sons, New York, NY, USA, 2011.

[8] V. Kolehmainen, M. Lassas, K. Niinimäki, and S. Siltanen, "Sparsity-promoting Bayesian inversion," *Inverse Problems*, vol. 28, no. 2, Article ID 025005, 2012.

[9] U. Grenander and M. I. Miller, "Representations of knowledge in complex systems," *Journal of the Royal Statistical Society*, vol. 56, no. 4, pp. 549–603, 1994.

[10] C. A. Zarzer, "On Tikhonov regularization with non-convex sparsity constraints," *Inverse Problems*, vol. 25, no. 2, Article ID 025006, 13 pages, 2009.

[11] D. L. Colton and R. Kress, *Integral Equation Methods in Scattering Theory*, vol. 57, John Wiley & Sons, New York, NY, USA, 1983.

[12] D. Colton and R. Kress, *Inverse Acoustic and Electromagnetic Scattering Theory*, vol. 93 of *Applied Mathematical Sciences*, Springer, Berlin, Germany, 2nd edition, 1998.

[13] G. Dalquist and A. Bjorck, *Numerical Methods in Scientific Computing*, vol. 102, SIAM, 2008.

[14] M. Chen, Q. Shao, and J. G. Ibrahim, *Monte Carlo methods in Bayesian Computation*, Springer Series in Statistics, Springer, New York, NY, USA, 2000.

[15] M. A. Pinsky, *Introduction to Fourier Analysis and Wavelets*, vol. 102 of *Graduate Studies in Mathematics*, American Mathematical Society, 2002.

[16] P. de Valpine, "Improved estimation of normalizing constants from Markov chain MONte Carlo output," *Journal of Computational and Graphical Statistics*, vol. 17, no. 2, pp. 333–351, 2008.

[17] B. Silverman, *Density Estimation for Statistics and Data Analysis*, Monographs on Statistics and Applied Probability no. 26, Chapman & Hall, New York, NY, USA, 1986.

Hindawi Publishing Corporation
Mathematical Problems in Engineering
Volume 2014, Article ID 137616, 13 pages
http://dx.doi.org/10.1155/2014/137616

Research Article

Approximate Sparsity and Nonlocal Total Variation Based Compressive MR Image Reconstruction

Chengzhi Deng, Shengqian Wang, Wei Tian, Zhaoming Wu, and Saifeng Hu

Department of Information Engineering, Nanchang Institute of Technology, Nanchang, China

Correspondence should be addressed to Chengzhi Deng; dengchengzhi@126.com

Received 27 March 2014; Revised 11 August 2014; Accepted 14 August 2014; Published 28 August 2014

Academic Editor: Fatih Yaman

Recent developments in compressive sensing (CS) show that it is possible to accurately reconstruct the magnetic resonance (MR) image from undersampled k-space data by solving nonsmooth convex optimization problems, which therefore significantly reduce the scanning time. In this paper, we propose a new MR image reconstruction method based on a compound regularization model associated with the nonlocal total variation (NLTV) and the wavelet approximate sparsity. Nonlocal total variation can restore periodic textures and local geometric information better than total variation. The wavelet approximate sparsity achieves more accurate sparse reconstruction than fixed wavelet ℓ_0 and ℓ_1 norm. Furthermore, a variable splitting and augmented Lagrangian algorithm is presented to solve the proposed minimization problem. Experimental results on MR image reconstruction demonstrate that the proposed method outperforms many existing MR image reconstruction methods both in quantitative and in visual quality assessment.

1. Introduction

Magnetic resonance imaging (MRI) is a noninvasive and nonionizing imaging processing. Due to its noninvasive manner and intuitive visualization of both anatomical structure and physiological function, MRI has been widely applied in clinical diagnosis. Imaging speed is important in many MRI applications. However, both scanning and reconstruction speed of MRI will affect the quality of reconstructed image. In spite of advances in hardware and pulse sequences, the speed, at which the data can be collected in MRI, is fundamentally limited by physical and physiological constraints. Therefore many researchers are seeking methods to reduce the amount of acquired data without degrading the image quality [1–3].

In recent years, the compressive sensing (CS) framework has been successfully used to reconstruct MR images from highly undersampled k-space data [4–9]. According to CS theory [10, 11], signals/images can be accurately recovered by using significantly fewer measurements than the number of unknowns or than mandated by traditional Nyquist sampling. MR image acquisition can be looked at as a special case of CS where the sampled linear combinations are simply individual Fourier coefficients (k-space samples). Therefore, CS is claimed to be able to make accurate reconstructions from a small subset of k-space data. In compressive sensing MRI (CSMRI), we can reconstruct a MR image with good quality from only a small number of measurements. Therefore, the application of CS to MRI has potential for significant scan time reductions, with benefits for patients and health care economics.

Because of the ill-posed nature of the CSMRI reconstruction problem, regularization terms are required for a reasonable solution. In existing CSMRI models, the most popular regularizers are ℓ_0, ℓ_1 sparsity [4, 9, 12] and total variation (TV) [3, 13]. The ℓ_0 sparsity regularized CSMRI model can be understood as a penalized least square with ℓ_0 norm penalty. It is well known that the complexity of this model is proportional with the number of variables. Particularly when the number is large, solving the model generally is intractable. The ℓ_1 regularization problem can be transformed into an equivalent convex quadratic optimization problem and, therefore, can be very efficiently solved. And under some conditions, the resultant solution of ℓ_1 regularization coincides with one of the solutions of ℓ_0 regularization [14]. Nevertheless, while ℓ_1 regularization provides the best convex approximation to ℓ_0 regularization

and it is computationally efficient, the ℓ_1 regularization often introduces extra bias in estimation and cannot reconstruct an image with the least measurements when applied to CSMRI [15]. In recent years, the ℓ_q $(0 < q < 1)$ regularization [16, 17] was introduced into CSMRI, since ℓ_q regularization can assuredly generate much sparser solutions than ℓ_1 regularization. Although the ℓ_q regularizations achieve better performance, they always fall into local minima. Moreover, which q should yield a best result is also a problem. Trzasko and Manduca [18] proposed a CSMRI paradigm based on homotopic approximation of the ℓ_0 quasinorm. Although this method has no guarantee of achieving a global minimum, it achieves accurate MR image reconstructions at higher undersampling rates than ℓ_1 regularization. And it was faster than those ℓ_q regularization methods. Recently, Chen and Huang [19] accelerated MRI by introducing the wavelet tree structural sparsity into the CSMRI.

Despite high effectiveness in CSMRI recovery, sparsity and TV regularizers often suffer from undesirable visual artifacts and staircase effects. To overcome those drawbacks, some hybrid sparsity and TV regularization methods [5–8] are proposed. In [5], Huang et al. proposed a new optimization algorithm for MR image reconstruction method, named fast composite splitting algorithm (FCSA), which is based on the combination of variable and operator splitting techniques. In [8], Yang et al. proposed a variable splitting method (RecPF) to solve hybrid sparsity and TV regularized MR image reconstruction optimization problem. Ma et al. [20] proposed an operator splitting algorithm (TVCMRI) for MR reconstruction. In order to deal with the problem of low and high frequency coefficients measurement, Zhang et al. [6] proposed a new so-called TVWL2-L1 model which measures low frequency coefficients and high frequency coefficients with ℓ_2 norm and ℓ_1 norm. In [7], an experimental study on the choice of CSMRI regularizations was given. Although the classical TV regularization performs well in CSMRI reconstruction while preserving edges, especially for cartoon-like MR images, it is well known that TV regularization is not suitable for images with fine details and it often tends to oversmooth image details and textures. Nonlocal TV regularization extends the classical TV regularization by nonlocal means filter [21] and has been shown to outperform the TV in several inverse problems such as image deonising [22], deconvolution [23], and compressive sensing [24, 25]. In order to improve the signal-to-noise ratio and preserve the fine details of MR images, Gopi et al. [26], Huang and Yang [27], and Liang et al. [28] have proposed nonlocal TV regularization based MR reconstruction and sensitivity encoding reconstruction.

In this paper, we proposed a novel compound regularization based compressive MR image reconstruction method, which exploits the nonlocal total variation (NLTV) and the approximate sparsity prior. The approximate sparsity, which is used to replace the traditional ℓ_0 regularizer and ℓ_1 regularizer of compressive MR image reconstruction model, is sparser and much easier to be solved. The NLTV is much better than TV for preserving the sharp edges and meanwhile recovering the local structure details. In order to compound regularization model, we develop an alternative iterative scheme by using the variable splitting and augmented Lagrangian algorithm. Experimental results show that the proposed method can effectively improve the quality of MR image reconstruction. The rest of the paper is organized as follows. In Section 2 we review the compressive sensing and MRI reconstruction. In Section 3 we propose our model and algorithm. The experimental results and conclusions will be shown in Sections 4 and 5, respectively.

2. Compressive Sensing and MRI Reconstruction

Compressive sensing [10, 11], as a new sampling and compression theory, is able to reconstruct an unknown signal from a very limited number of samples. It provides a firm theoretical foundation for the accurate reconstruction of MRI from highly undersampled K-space measurements and significantly reduces the MRI scan duration.

Suppose $\mathbf{u} \in \Re^N$ is a MR image and $\mathbf{F} \in \Re^{M\times N}$ is a partial Fourier transform; then the sampling measurement $\mathbf{b} \in \Re^M$ of MR image $\mathbf{u}$ in K-space can be defined as

$$\mathbf{b} = \mathbf{F}\mathbf{u}. \tag{1}$$

The compressive MR image reconstruction problem is to recover $\mathbf{u}$ given the measurement $\mathbf{b}$ and the sampling matrix $\mathbf{F}$. Undersampling occurs whenever the number of K-space sample is less than the number of unknowns $(M < N)$. In that case, the compressive MR image reconstruction is an underdetermined problem.

In general, compressive sensing reconstructs the unknowns $\mathbf{u}$ from the measurements $\mathbf{b}$ by minimizing the ℓ_0 norm of the sparsified image $\mathbf{\Phi u}$, where $\mathbf{\Phi}$ represents a sparsity transform for the image. In this paper, we choose the orthonormal wavelet transform as the sparsity transform for the image. Then the typical compressive MR image reconstruction is obtained by solving the following constrained optimization problem [4, 9, 12]:

$$\begin{aligned} &\min_{\mathbf{u}} \|\mathbf{\Phi u}\|_0 \\ &\text{s.t.} \quad \mathbf{b} = \mathbf{F}\mathbf{u}. \end{aligned} \tag{2}$$

However, in terms of computational complexity, the ℓ_0 norm optimization problem (2) is a typical NP-hard problem, and it was difficult to solve. According to the certain condition of the restricted isometric property, the ℓ_0 norm can be replaced by the ℓ_1 norm. Therefore, the optimization problem (2) is relaxed to alternative convex optimization problem as follows:

$$\begin{aligned} &\min_{\mathbf{u}} \|\mathbf{\Phi u}\|_1 \\ &\text{s.t.} \quad \mathbf{b} = \mathbf{F}\mathbf{u}. \end{aligned} \tag{3}$$

When the measurements $\mathbf{b}$ are contaminated with noise, the typical compressive MR image reconstruction problem using ℓ_1 relaxation of the ℓ_0 norm is formulated as the following unconstrained Lagrangian version:

$$\min_{\mathbf{u}} \|\mathbf{F}\mathbf{u} - \mathbf{b}\|_2^2 + \alpha\|\mathbf{\Phi u}\|_1, \tag{4}$$

where α is a positive parameter.

Despite high effectiveness of sparsity regularized compressive MR image reconstruction methods, they often suffer from undesirable visual artifacts such as Gibbs ringing in the result. Due to its desirable ability to preserve edges, total variation (TV) model is successfully used in compressive MR image reconstruction [3, 13]. But the TV regularizer has still some limitations that restrict its performance, which cannot generate good enough results for images with many small structures and often suffers from staircase artifacts. In order to combine the advantages of sparsity-based and TV model and avoid their main drawbacks, a TV regularizer, corresponding to a finite-difference for the sparsifying transform, is typically incorporated into the sparsity regularized compressive MR image reconstruction [5–8]. In this case the optimization problem is written as

$$\min_{\mathbf{u}} \|\mathbf{F}\mathbf{u} - \mathbf{b}\|_2^2 + \alpha\|\mathbf{\Phi}\mathbf{u}\|_1 + \beta\|\nabla\mathbf{u}\|_1, \tag{5}$$

where β is a positive parameter. The TV was defined discretely as $\|\nabla\mathbf{u}\|_1 = \sum_i(|\nabla_h u_i| + |\nabla_v u_i|)$, where ∇_h and ∇_v are the horizontal and the vertical gradient operators, respectively. The compound optimization model (5) is based on the fact that the piecewise smooth MR images can be sparsely represented by the wavelet and should have small total variations.

3. Proposed Model and Algorithm

As mentioned above, joint TV and ℓ_1 norm minimization model is a useful way to reconstruct MR images. However, they have still some limitations that restrict their performance. ℓ_0 norm needs a combinatorial search for its minimization and its too high sensibility to noise. ℓ_1 problems can be very efficiently solved. But the solution is not sparse, which influences the performance of MRI reconstruction. The TV model can preserve edges, but it tends to flatten inhomogeneous areas, such as textures. To overcome those shortcomings, a novel method is proposed for compressive MR imaging based on the wavelet approximate sparsity and nonlocal total variation (NLTV) regularization, named WasNLTV.

3.1. Approximate Sparsity. The problems of using ℓ_0 norm in compressive MR imaging (i.e., the need for a combinatorial search for its minimization and its too high sensibility to noise) are both due to the fact that the ℓ_0 norm of a vector is a discontinuous function of that vector. The same as [29, 30], our idea is to approximate this discontinuous function by a continuous one, named approximate sparsity function, which provides smooth measure of ℓ_0 norm and better sparsity than ℓ_1 regularizer.

The approximate sparsity function is defined as

$$\psi_\sigma(x) = \frac{2}{\pi}\arctan\left(\frac{|x|}{\sigma^2}\right), \quad x \in \mathfrak{R},\ \sigma \in \mathfrak{R}^+. \tag{6}$$

The parameter σ may be used to control the accuracy with which ψ_σ approximate the Kronecker delta. In mathematical terms, we have

$$\lim_{\sigma\to 0}\psi_\sigma(x) = \begin{cases} 1, & x \neq 0, \\ 0, & x = 0. \end{cases} \tag{7}$$

Define the continuous multivariate approximate sparsity function $\Psi_\sigma(\mathbf{x})$ as

$$\Psi_\sigma(\mathbf{x}) = \sum_{i=1}^{m}\psi_\sigma(x_i), \quad \mathbf{x} \in \mathfrak{R}^{m\times 1}. \tag{8}$$

It is clear from (7) that $\Psi_\sigma(\mathbf{x})$ is an indicator of the number of zero-entries in $\mathbf{x}$ for small values of σ. Therefore, ℓ_0 norm can be approximate by

$$\|\mathbf{x}\|_0 \approx \Psi_\sigma(\mathbf{x}) = \sum_{i=1}^{m}\psi_\sigma(x_i). \tag{9}$$

Note that the larger the value of σ, the smoother the $\Psi_\sigma(\mathbf{x})$ and the worse the approximation to ℓ_0 norm; the smaller the value of ℓ_0 norm, the closer the behavior of $\Psi_\sigma(\mathbf{x})$ to ℓ_0 norm.

3.2. Nonlocal Total Variation. Although the classical TV is surprisingly efficient for preserving edges, it is well known that TV is not suitable for images with fine structures, details, and textures which are very important to MR images. The NLTV is a variational extension of the nonlocal means filter proposed by Wang et al. [30]. NLTV uses the whole image information instead of using adjacent pixel information to calculate the gradients in regularization term. The NLTV has been proven to be more efficient than TV for improving the signal-to-noise ratio, on preserving not only sharp edges, but also fine details and repetitive patterns [26–28]. In this paper, we use the NLTV to replace the TV in compound regularization based compressive MR image reconstruction.

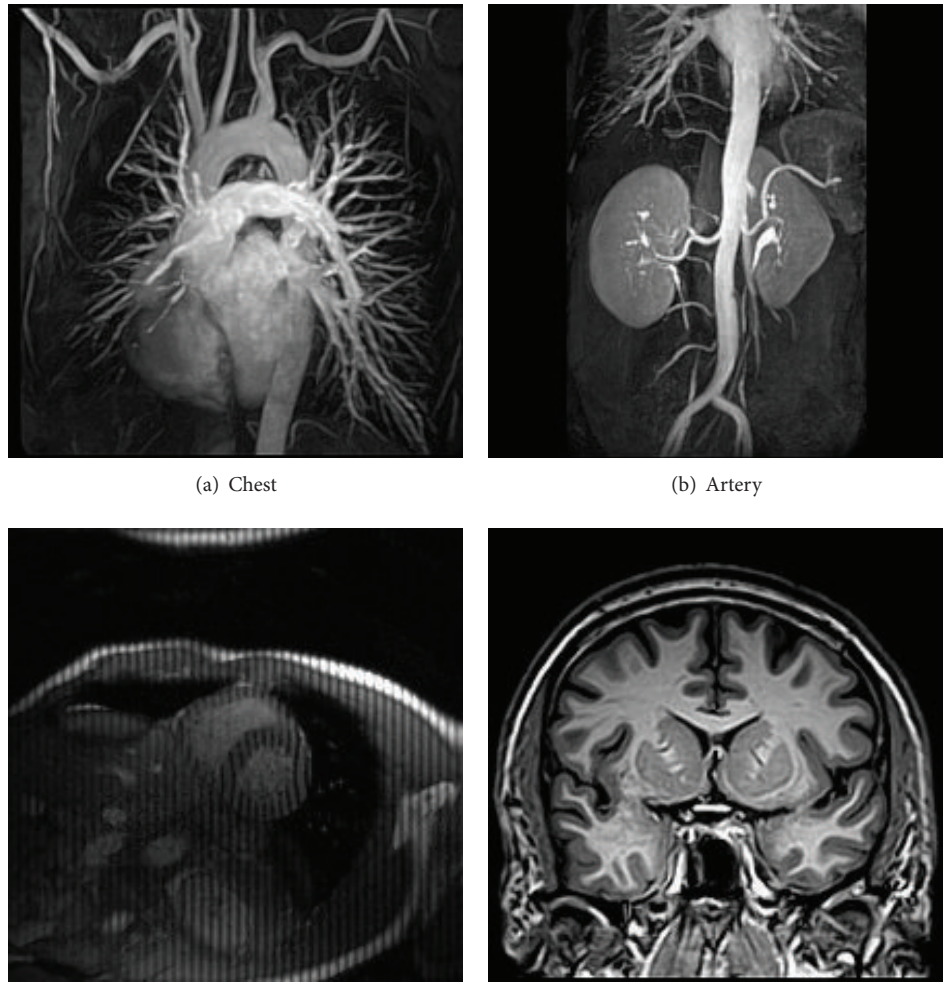
Let $\Omega \subset \mathfrak{R}^2$, $i, j \in \Omega$, $u(x)$ be a real function $u : \Omega \to \mathfrak{R}$, and let $w(x, y)$ be a weight function. For a given image $u(x)$, the weighted graph gradient is $\nabla_{NL}u(x)$ if defined as the vector of all directional derivatives $\nabla_{NL}u(x, \cdot)$ at x:

$$\nabla_{NL}u(x, y) := (u(y) - u(x))\sqrt{w(x, y)}, \quad \forall y \in \Omega. \tag{10}$$

The directional derivatives apply to all the nodes y since the weight $w(x, y)$ is extended to the whole domain $\Omega\times\Omega$. Let us denote vectors such that $\overrightarrow{p} = p(x, y) \in \Omega \times \Omega$; the nonlocal graph divergence $(\mathrm{div}_{NL}\overrightarrow{p}) : \Omega \times \Omega \to \Omega$ is defined as the adjoint of the nonlocal gradient:

$$\left(\mathrm{div}_{NL}\overrightarrow{p}\right)(x) := \sum_{y\in\Omega}(p(x, y) - p(y, x))\sqrt{w(x, y)}. \tag{11}$$

Due to being analogous to classical TV, the ℓ_1 norm is in general more efficient than the ℓ_2 norm for sparse



(a) Chest (b) Artery

(c) Cardiac (d) Brain

Figure 1: 2D test MR images.

reconstruction. In this paper, we are interested in NLTV. Based on the above definition, the NLTV is defined as follows:

$$\begin{aligned}\left\|\nabla_{NL}\mathbf{u}\right\|_1 &:= \sum_{x\in\Omega}\left|\nabla_{NL}u\left(x\right)\right| \\ &= \sum_{x\in\Omega}\sqrt{\sum_{y\in\Omega}\left(u\left(x\right)-u\left(y\right)\right)^2 w\left(x,y\right)}.\end{aligned} \tag{12}$$

The weight function $w(x, y)$ denotes how much the difference between pixels x and y is penalized in the images, which is calculated by

$$w\left(x,y\right)=\frac{1}{C_u}\exp\left(\frac{-\left\|f_u\left(x\right)-f_u\left(y\right)\right\|^2}{2h^2}\right), \tag{13}$$

where $f_u(x)$ and $f_u(y)$ denote a small patch in image u centering at the coordinates x and y, respectively. $C_u = \sum_{y\in\Omega} w(x, y)$ is the normalizing factor. h is a filtering parameter.

3.3. The Description of Proposed Model and Algorithm. According to the compressive MR image reconstruction models described in Section 2, the proposed WasNLTV model for compressive MR image reconstruction is

$$\min_{\mathbf{u}}\left\|\mathbf{F}\mathbf{u}-\mathbf{b}\right\|_2^2+\alpha\Psi\left(\mathbf{\Phi}\mathbf{u}\right)+\beta\left\|\nabla_{NL}\mathbf{u}\right\|_1. \tag{14}$$

It should be noted that the optimization problem in (14), although convex, is very hard to solve owing to nonsmooth terms and its huge dimensionality. To solve the problem in (14), we use the variable splitting and augmented Lagrangian algorithm following closely the methodology introduced in [31]. The core idea is to introduce a set of new variables per regularizer and then exploit the alternating direction method of multipliers (ADMM) to solve the resulting constrained optimization problems.

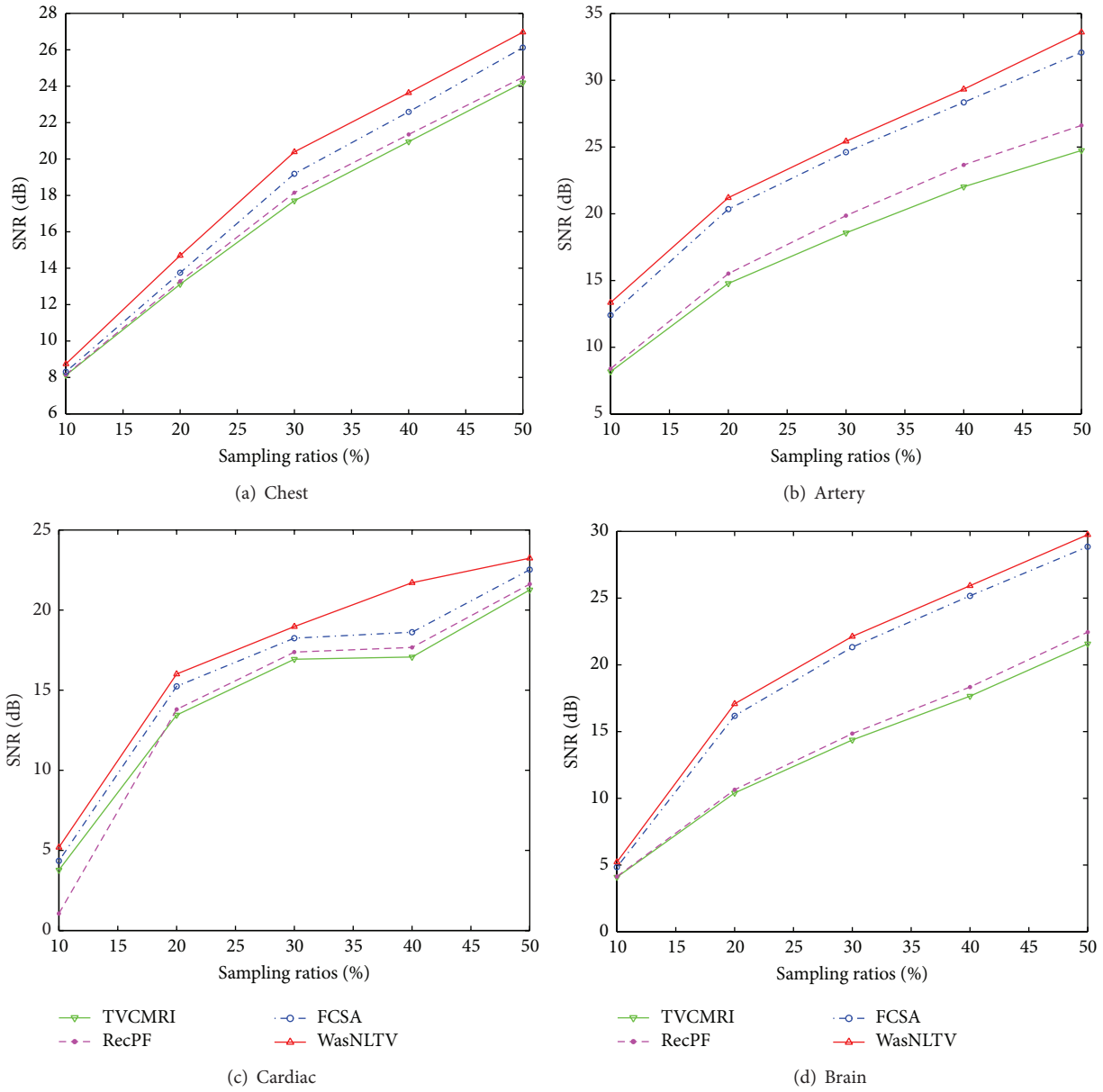


Figure 2: Performance comparisons (sampling rate versus SNR) on different MR images.

By introducing an intermediate variable vector $(\mathbf{v}_1, \mathbf{v}_2, \mathbf{v}_3)$, the problem (14) can be transformed into an equivalent one; that is,

$$\min_{\mathbf{u},\mathbf{v}_1,\mathbf{v}_2,\mathbf{v}_3} \|\mathbf{v}_1 - \mathbf{b}\|_2^2 + \alpha\Psi(\mathbf{v}_2) + \beta\|\mathbf{v}_3\|_1$$
$$\text{s.t.} \quad \mathbf{F}\mathbf{u} = \mathbf{v}_1, \quad \mathbf{\Phi}\mathbf{u} = \mathbf{v}_2, \quad \nabla_{NL}\mathbf{u} = \mathbf{v}_3. \tag{15}$$

The optimization problem (15) can be written in a compact form as follows:

$$\min_{\mathbf{u},\mathbf{v}} g(\mathbf{v})$$
$$\text{s.t.} \quad \mathbf{G}\mathbf{u} + \mathbf{B}\mathbf{v} = 0, \tag{16}$$

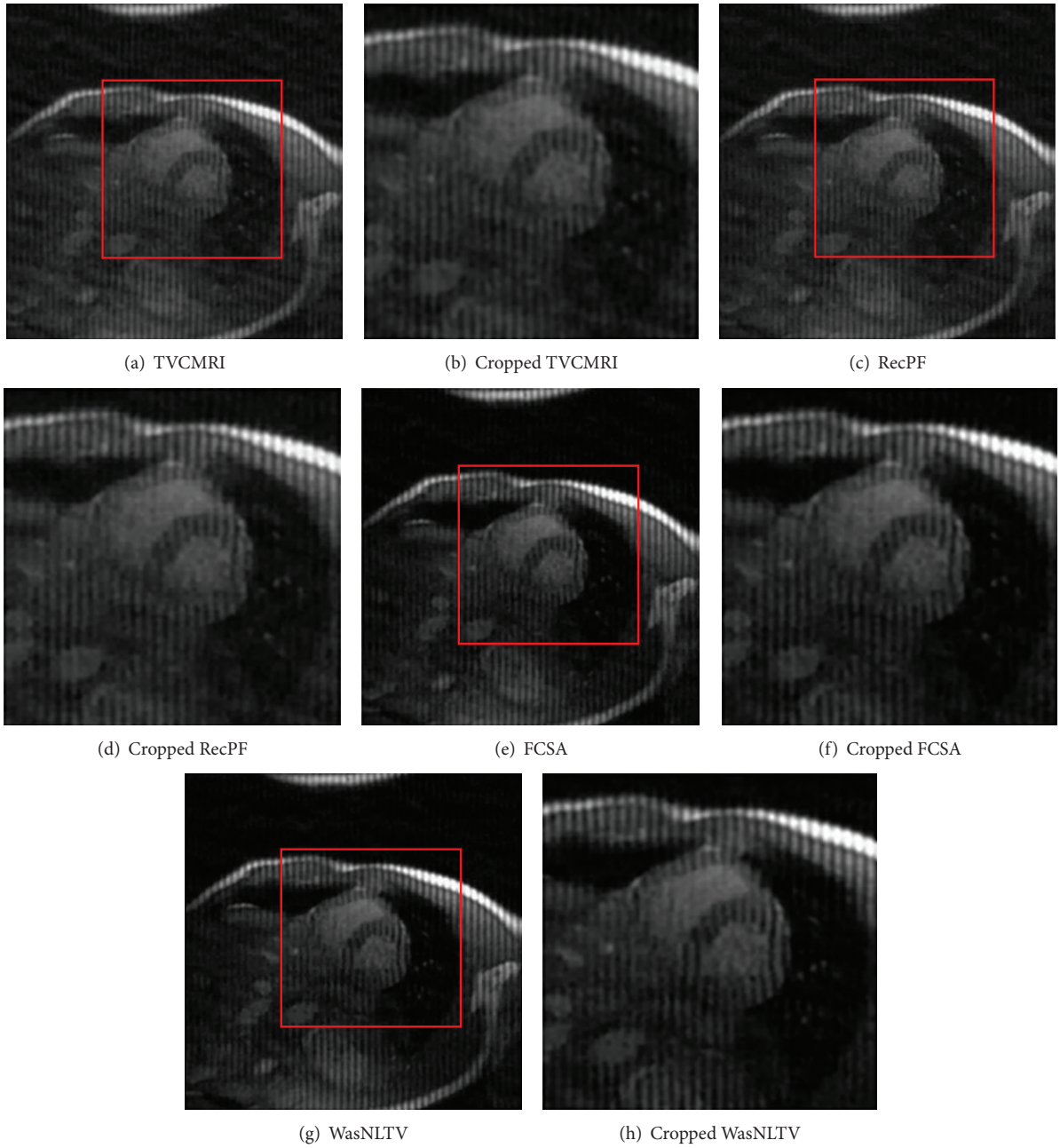
where

$$\mathbf{v} \equiv (\mathbf{v}_1, \mathbf{v}_2, \mathbf{v}_3),$$
$$g(\mathbf{v}) = \|\mathbf{v}_1 - \mathbf{b}\|_2^2 + \alpha\Psi(\mathbf{v}_2) + \beta\|\mathbf{v}_3\|_1,$$
$$\mathbf{G} = \begin{bmatrix} \mathbf{F} \\ \mathbf{\Phi} \\ \nabla_{NL} \end{bmatrix}, \qquad \mathbf{B} = \begin{bmatrix} -\mathbf{I} & 0 & 0 \\ 0 & -\mathbf{I} & 0 \\ 0 & 0 & -\mathbf{I} \end{bmatrix}. \tag{17}$$

The augmented Lagrangian of problem (16) is

$$\mathscr{L}(\mathbf{u}, \mathbf{v}, \mathbf{d}) = g(\mathbf{v}) + \frac{\mu}{2}\|\mathbf{G}\mathbf{u} + \mathbf{B}\mathbf{v} - \mathbf{d}\|_2^2, \tag{18}$$

where $\mu > 0$ is a positive constant, $\mathbf{d} \equiv (\mathbf{d}_1, \mathbf{d}_2, \mathbf{d}_3)$, and $\mathbf{d}/\mu$ denotes the Lagrangian multipliers associated to



(a) TVCMRI (b) Cropped TVCMRI (c) RecPF

(d) Cropped RecPF (e) FCSA (f) Cropped FCSA

(g) WasNLTV (h) Cropped WasNLTV

Figure 3: Reconstructed cardiac MR images from 20% sampling.

the constraint $\mathbf{Gu} + \mathbf{Bv} = 0$. The basic idea of the augmented Lagrangian method is to seek a saddle point of $\mathscr{L}(\mathbf{u}, \mathbf{v}, \mathbf{d})$, which is also the solution of problem (16). By using ADMM algorithm, we solve the problem (16) by iteratively solving the following problems:

$$\left(\mathbf{u}^{k+1}, \mathbf{v}^{k+1}\right) = \min_{\mathbf{u},\mathbf{v}} \mathscr{L}\left(\mathbf{u}, \mathbf{v}, \mathbf{d}\right), \tag{19}$$

$$\mathbf{d}^{k+1} = \mathbf{d}^{k} - \mathbf{G}\mathbf{u}^{k+1} - \mathbf{B}\mathbf{v}^{k+1}. \tag{20}$$

It is evident that the minimization problem (19) is still hard to solve efficiently in a direct way, since it involves a nonseparable quadratic term and nondifferentiability terms. To solve this problem, a quite useful ADMM algorithm is employed, which alternatively minimizes one variable while fixing the other variables. By using ADMM, the problem (19) can be solved by the following four subproblems with respect to $\mathbf{u}$ and $\mathbf{v}$.

(1) $\mathbf{u}$ subproblem: by fixing $\mathbf{v}$ and $\mathbf{d}$, the optimization problem (19) to be solved is

$$\begin{aligned}\mathbf{u}^{k+1} = \min_{\mathbf{u}} &\frac{\mu}{2}\left\|\mathbf{F}\mathbf{u} - \mathbf{v}_1^k - \mathbf{d}_1^k\right\|_2^2 + \frac{\mu}{2}\left\|\mathbf{\Phi}\mathbf{u} - \mathbf{v}_2^k - \mathbf{d}_2^k\right\|_2^2 \\ &+ \frac{\mu}{2}\left\|\nabla_{NL}\mathbf{u} - \mathbf{v}_3^k - \mathbf{d}_3^k\right\|_2^2.\end{aligned} \tag{21}$$

Initialization: set $k = 0$, choose α, β, μ, $\mathbf{u}^0$, $\mathbf{v}_1^0$, $\mathbf{v}_2^0$, and $\mathbf{v}_3^0$
repeat:
  compute u sub-problem: $\mathbf{u}^{k+1} = \min_{\mathbf{u}} \mu/2\|\mathbf{Gu} + \mathbf{Bv} - \mathbf{d}\|_2^2$ based on (23)
  for $i = 1, \ldots, 3$
    computer $\mathbf{v}_i$ **sub-problem**: $\mathbf{v}_i^{k+1} = \min_{\mathbf{v}_i} g(\mathbf{v}_i) + \mu/2\|\mathbf{Gu} + \mathbf{Bv}_i - \mathbf{d}_i\|_2^2$ based on (25), (27) and (29)
  end for
  update Lagrange multipliers:
    $\mathbf{d}_1^{k+1} = \mathbf{d}_1^k - \mathbf{Fu}^{k+1} + \mathbf{v}_1^{k+1}$
    $\mathbf{d}_2^{k+1} = \mathbf{d}_2^k - \mathbf{\Phi u}^{k+1} + \mathbf{v}_2^{k+1}$
    $\mathbf{d}_3^{k+1} = \mathbf{d}_3^k - \nabla_{NL}\mathbf{u}^{k+1} + \mathbf{v}_3^{k+1}$
  update iteration: $k = k + 1$
until the stopping criterion is satisfied.

Algorithm 1: Pseudocode of WasNLTV based compressive MR image reconstruction.

It is clear that problem (21) is a quadratic function. By direct computation, we get the Euler-Lagrange equation for (21):

$$\mu\mathbf{F}^T\left(\mathbf{Fu} - \mathbf{v}_1^k - \mathbf{d}_1^k\right) + \mu\mathbf{\Phi}^T\left(\mathbf{\Phi u} - \mathbf{v}_2^k - \mathbf{d}_2^k\right) + \mu\mathrm{div}_{NL}\left(\nabla_{NL}\mathbf{u} - \mathbf{v}_3^k - \mathbf{d}_3^k\right) = 0. \tag{22}$$

Therefore, the solution of problem (21) is

$$\mathbf{u}^{k+1} = \left(\mathbf{F}^T\mathbf{F} + \mathbf{\Phi}^T\mathbf{\Phi} + \Delta_{NL}\right)^{-1} \times \left(\mathbf{F}^T\left(\mathbf{v}_1^{k+1} + \mathbf{d}_1^{k+1}\right) + \mathbf{\Phi}^T\left(\mathbf{v}_2^{k+1} + \mathbf{d}_2^{k+1}\right) + \mathrm{div}_{NL}\left(\mathbf{v}_3^{k+} + \mathbf{d}_3^{k+1}\right)\right). \tag{23}$$

Due to the computational complexity of NLTV, the same as [27], the NLTV regularization in this paper only runs one time.

(2) $\mathbf{v}_1$ subproblem: by fixing $\mathbf{v}_2$, $\mathbf{v}_3$, $\mathbf{u}$, and $\mathbf{d}$, the optimization problem (19) to be solved is

$$\mathbf{v}_1^{k+1} = \min_{\mathbf{v}_1}\left\|\mathbf{v}_1 - \mathbf{b}\right\|_2^2 + \frac{\mu}{2}\left\|\mathbf{Fu}^{k+1} - \mathbf{v}_1 - \mathbf{d}_1^{k+1}\right\|_2^2. \tag{24}$$

Clearly, the problem (24) is a quadratic function; its solution is simply

$$\mathbf{v}_1^{k+1} = \frac{\left(2\mathbf{b} + \mu\mathbf{Fu}^{k+1} - \mu\mathbf{d}_1^{k+1}\right)}{(2+\mu)}. \tag{25}$$

(3) $\mathbf{v}_2$ subproblem: by fixing $\mathbf{v}_1$, $\mathbf{v}_3$, $\mathbf{u}$, and $\mathbf{d}$, the optimization problem (19) to be solved is

$$\mathbf{v}_2^{k+1} = \min_{\mathbf{v}_2}\alpha\Psi(\mathbf{v}_2) + \frac{\mu}{2}\left\|\mathbf{\Phi u}^{k+1} - \mathbf{v}_2 - \mathbf{d}_2^{k+1}\right\|_2^2. \tag{26}$$

The same as problem (24), the problem (26) is a quadratic function and its gradient $\nabla_{\mathbf{v}_2}$ is simplified as $\nabla_{\mathbf{v}_2} = \mu(\mathbf{v}_2 - \mathbf{\Phi u}^{k+1} + \mathbf{d}_2^{k+1}) + 2\alpha\sigma^2/\pi(\sigma^2 + \mathbf{v}_2)^{-2}$. The steepest descent method is desirable to use to solve (26) iteratively by applying

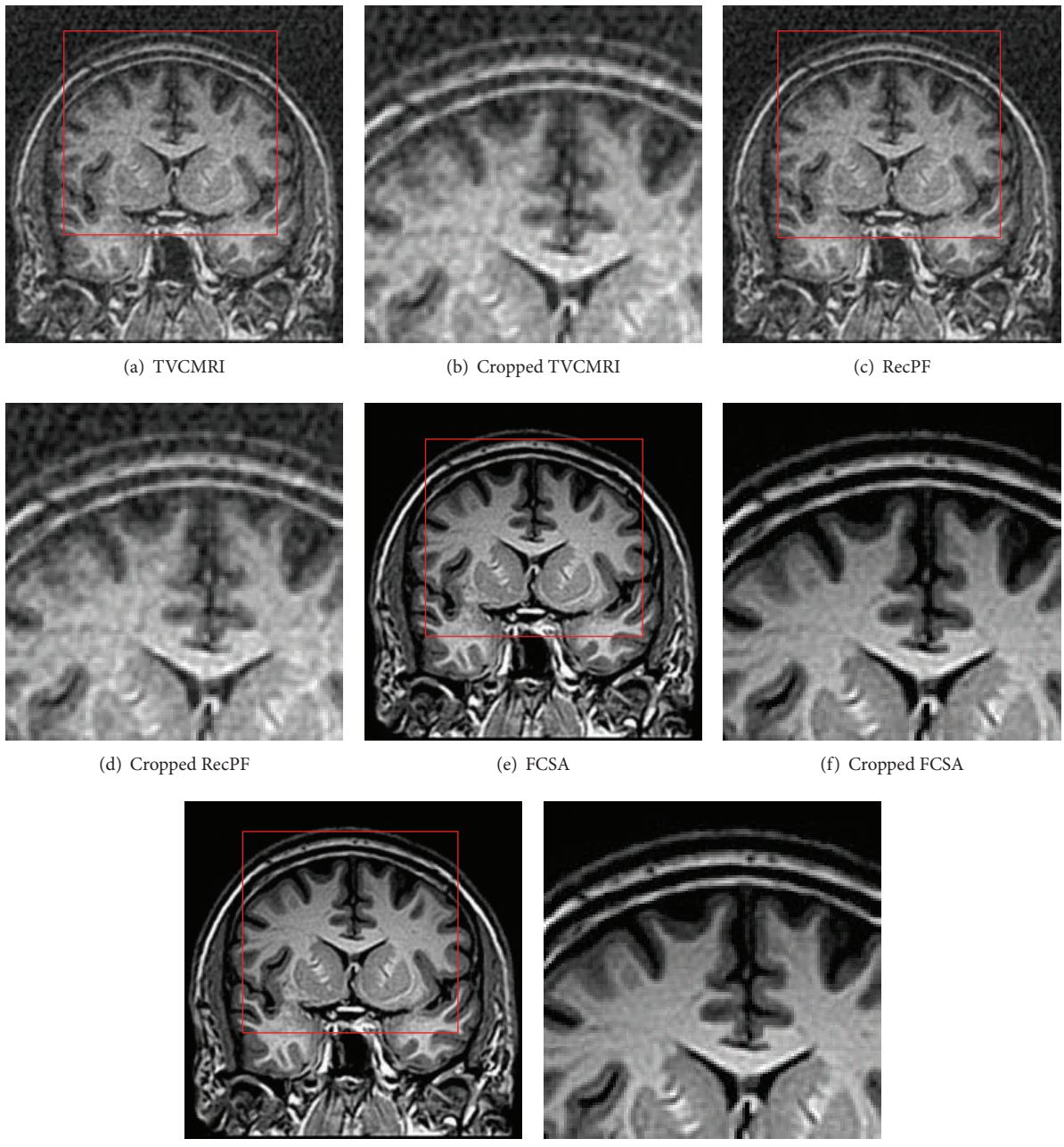
$$\mathbf{v}_2^{k+1} = \mathbf{v}_2^k - \eta\nabla_{\mathbf{v}_2}. \tag{27}$$

(4) $\mathbf{v}_3$ subproblem: by fixing $\mathbf{v}_1$, $\mathbf{v}_2$, $\mathbf{u}$, and $\mathbf{d}$, the optimization problem (19) to be solved is

$$\mathbf{v}_3^{k+1} = \min_{\mathbf{v}_3}\beta\left\|\mathbf{v}_3\right\|_1 + \frac{\mu}{2}\left\|\nabla_{NL}\mathbf{u}^{k+1} - \mathbf{v}_3 - \mathbf{d}_3^{k+1}\right\|_2^2. \tag{28}$$

Table 1: SNR (dB) results of different methods with different sampling ratios.

Image	Samp. ratio	TVCMRI	RecPF	FCSA	WasNLTV
Chest (220 × 220)	10	8.12	8.15	8.31	8.74
	20	13.12	13.28	13.75	14.70
	30	17.71	18.15	19.19	20.39
	40	20.95	21.35	22.59	23.64
	50	24.19	24.49	26.12	26.97
Artery (220 × 220)	10	8.17	8.39	12.40	13.35
	20	14.78	15.51	20.35	21.20
	30	18.57	19.86	24.61	25.43
	40	22.02	23.66	28.35	29.33
	50	24.74	26.62	32.08	33.60
Cardiac (192 × 192)	10	3.77	1.06	4.34	5.19
	20	13.45	13.80	15.24	16.01
	30	16.93	17.38	18.25	18.97
	40	17.07	17.67	18.62	21.71
	50	21.26	21.62	22.52	23.24
Brain (210 × 210)	10	4.09	4.14	4.84	5.23
	20	10.40	10.64	16.18	17.58
	30	14.37	14.85	21.33	22.13
	40	17.65	18.34	25.17	25.93
	50	21.57	22.45	28.85	29.75



(a) TVCMRI (b) Cropped TVCMRI (c) RecPF

(d) Cropped RecPF (e) FCSA (f) Cropped FCSA

(g) WasNLTV (h) Cropped WasNLTV

Figure 4: Reconstructed brain MR images from 20% sampling.

Problem (28) is a ℓ_1 norm regularized optimization problem. Its solution is the well-known soft threshold [32]:

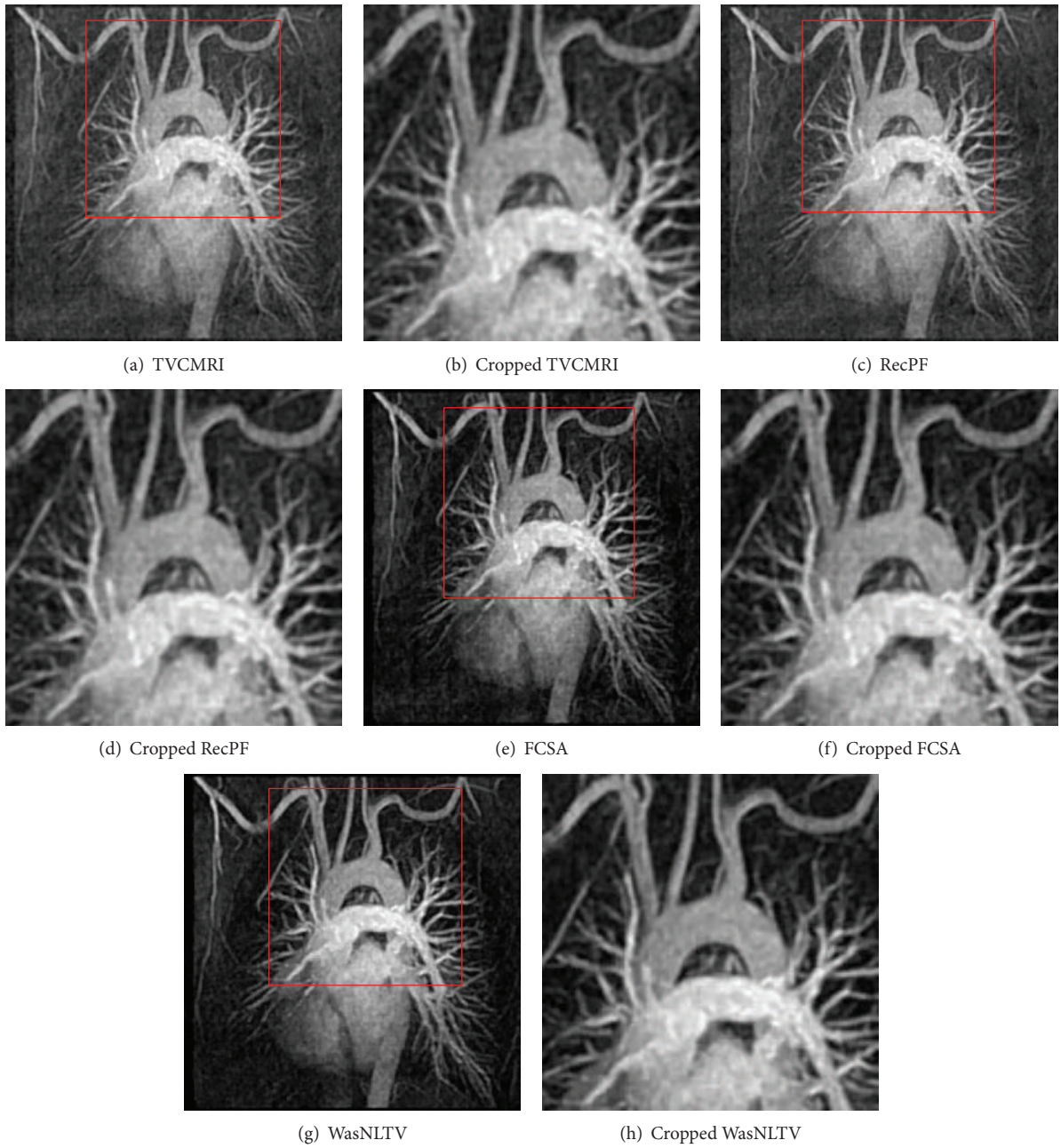
$$\mathbf{v}_3^{k+1} = \text{soft}\left(\nabla_{NL}\mathbf{u}^{k+1} - \mathbf{d}_3^{k+1}, \frac{\beta}{\mu}\right), \tag{29}$$

where $\text{soft}(y,\tau) = \text{sign}(y)\max\{|y| - \tau, 0\}$ denotes the component-wise application of soft-threshold function.

In conclusion, the ADMM algorithm for optimization problem (16) is shown in Algorithm 1.

4. Experimental Results

In this section, a series of experiments on four 2D MR images (named brain, chest, artery, and cardiac) are implemented to evaluate the proposed and existing methods. Figure 1 shows the test images. All experiments are conducted on a PC with an Intel Core i7-3520M, 2.90 GHz CPU, in MATLAB environment. The proposed method (named WasNLTV) is compared with the existing methods including TVCMRI [19], RecPF [8], and FCSA [5]. We evaluate the performance of various methods both visually and qualitatively in



(a) TVCMRI (b) Cropped TVCMRI (c) RecPF

(d) Cropped RecPF (e) FCSA (f) Cropped FCSA

(g) WasNLTV (h) Cropped WasNLTV

Figure 5: Reconstructed chest MR images from 20% sampling.

signal-to-noise ratio (SNR) and root-mean-square error (RMSE) values. The SNR and RMSE are defined as

$$\begin{aligned} \text{SNR} &= 10\log_{10}\frac{\|\mathbf{u}-E(\mathbf{u})\|_2^2}{\|\mathbf{u}-\tilde{\mathbf{u}}\|_2^2}, \\ \text{RMSE} &= \sqrt{E\left((\mathbf{u}-\tilde{\mathbf{u}})^2\right)}, \end{aligned} \tag{30}$$

where $\mathbf{u}$ and $\tilde{\mathbf{u}}$ denote the original image and the reconstructed image, respectively, and $E(\cdot)$ is the mean function.

For fair comparisons, experiment uses the same observation methods with TVCMRI. In the K-space, we randomly obtain more samples in low frequencies and fewer samples in higher frequencies. This sampling scheme is widely used for compressed MR image reconstructions. Suppose a MR image $\mathbf{u}$ has N pixels and the partial Fourier transform $\mathbf{F}$ in problem (1) consists of M rows of $N \times N$ matrix corresponding to the full 2D discrete Fourier transform. The M chosen rows correspond to the sampling measurements $\mathbf{b}$. Therefore, the sampling ratio is defined as M/N. In the experiments, the Gaussian white noise generated by $\sigma_n \times \text{randn}(M, 1)$ in MATLAB is added, where standard deviation $\sigma_n = 0.01$. The regularization parameters α, β, and μ are set as 0.001, 0.035, and 1, respectively. To be fair to compare the reconstruction

(a) TVCMRI (b) Cropped TVCMRI (c) RecPF

(d) Cropped RecPF (e) FCSA (f) Cropped FCSA

(g) WasNLTV (h) Cropped WasNLTV

Figure 6: Reconstructed artery MR images from 20% sampling.

MR images of various algorithms, all methods run 50 iterations and the Rice wavelet toolbox is used as the wavelet transform.

Table 1 summarizes the average reconstruction accuracy obtained by using different methods at different sampling ratios on the set of test images. From Table 1, it can be seen that the proposed WasNLTV method attains the highest SNR (dB) in all cases. Figure 2 plots the SNR values with sampling ratios for different images. It can also be seen that the WasNLTV method achieves the larger improvement of SNR values.

Table 2 gives the RMSE results of reconstructed MRI after applying different algorithms. From Table 2, it can be seen that WasNLTV method attains the lowest RMSE in all cases. As is known, the lower the RMSE is, the better the reconstructed image is. That is to say the MR images reconstructed by WasNLTV have the best visual quality.

To illustrate visual quality, reconstructed compressive MR images obtained using different methods with sampling ratios 20% are shown in Figures 3, 4, 5, and 6. For better visual comparison, we zoom in a small patch where the edge and texture are much more abundant. From the figures, it can be

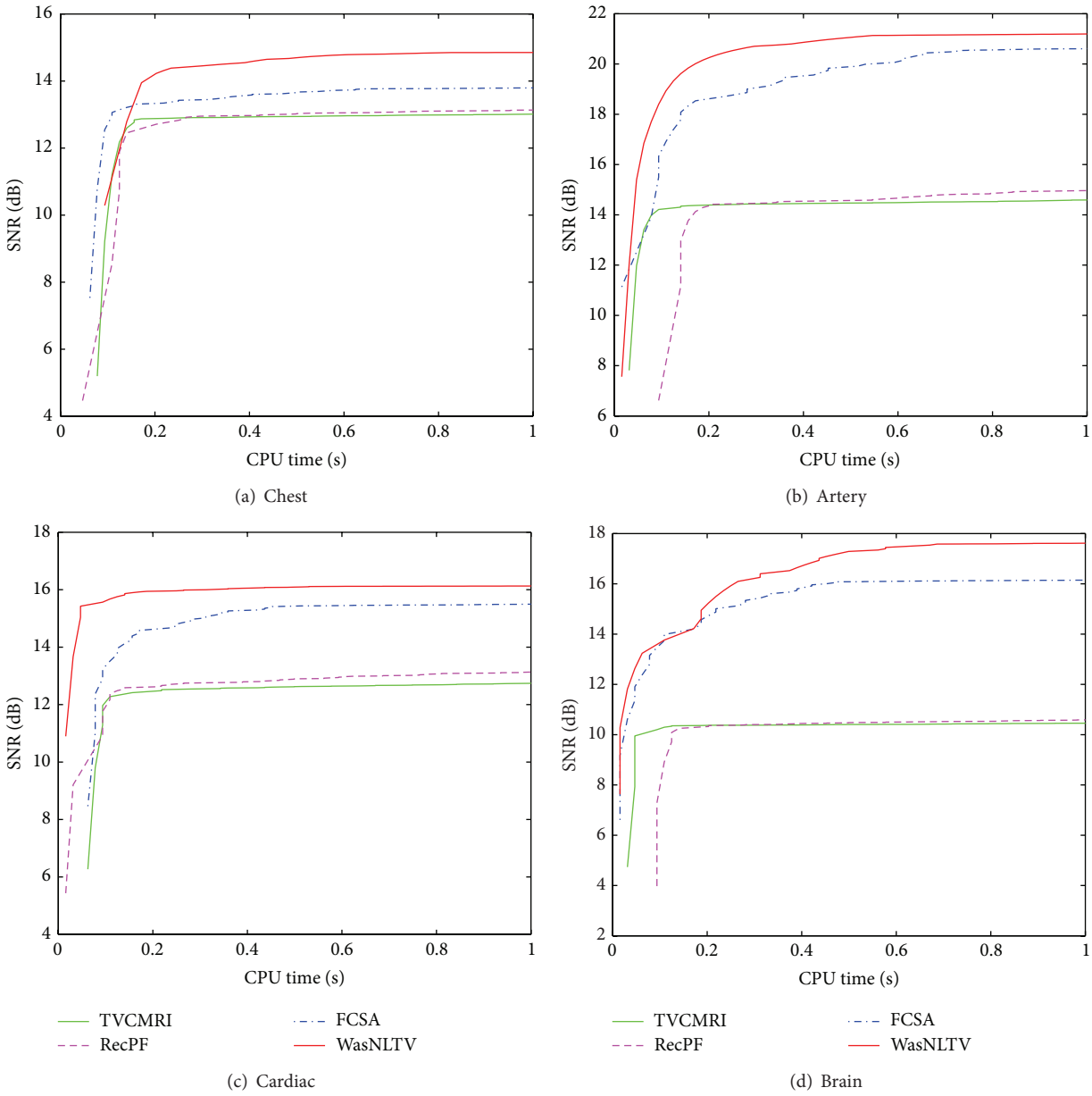


Figure 7: Performance comparisons (CPU-time versus SNR) on different MR images.

observed that the WasNLTV always obtains the best visual effects on all MR images. In particular, the edge of organs and tissues obtained by WasNLTV are much more clear and easy to identify.

Figure 7 gives the performance comparisons between different methods with sampling ratios 20% in terms of the CPU time over the SNR. In general, the computational complexity of NLTV is much higher than TV. In order to reduce the computational complexity of WasNLTV, in the experiment, we perform the NLTV regularization once in some iterations. Despite the higher computational complexity of WasNLTV, the WasNLTV obtains the best reconstruction results on all MR images by achieving the highest SNR in less CPU time.

5. Conclusions

In this paper, we propose a new compound regularization based compressive sensing MRI reconstruction model, which exploits the NLTV regularization and wavelet approximate sparsity prior. The approximate sparsity prior is used in compressive MR image reconstruction model instead of ℓ_0 or ℓ_1 norm, which can produce much sparser results. And the optimization problem is much easier to be solved. Because the NLTV takes advantage of the redundancy and self-similarity in a MR image, it can effectively avoid blocky artifacts caused by traditional TV regularization and keep fine edge of organs and tissues. As for the algorithm, we apply the variable splitting and augmented Lagrangian algorithm to

Table 2: RMSE results of different methods with different sampling ratios.

Image	Samp. ratio	TVCMRI	RecPF	FCSA	WasNLTV
Chest (220 × 220)	10	38.13	38.04	35.09	19.81
	20	10.90	10.60	9.71	8.67
	30	6.02	5.73	4.94	3.52
	40	3.72	3.58	3.00	2.20
	50	2.22	2.27	1.90	1.22
Artery (220 × 220)	10	43.95	43.81	41.03	14.67
	20	10.19	9.72	3.77	3.35
	30	5.06	4.46	2.18	1.77
	40	3.37	2.88	1.52	1.17
	50	2.17	1.88	1.09	0.64
Cardiac (192 × 192)	10	18.95	18.81	15.86	14.77
	20	8.12	7.89	5.71	4.28
	30	2.85	2.81	2.61	2.24
	40	2.16	2.20	2.15	1.74
	50	1.29	1.57	1.51	1.09
Brain (210 × 210)	10	40.27	40.11	35.87	29.11
	20	16.11	15.71	6.48	5.87
	30	9.48	9.05	3.63	3.14
	40	6.46	6.05	2.73	2.57
	50	3.25	3.07	2.41	2.01

solve the compound regularization minimization problem. Experiments on test images demonstrate that the proposed method leads to high SNR measure and more importantly preserves the details and edges of MR images.

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 anonymous referees for their valuable and helpful comments. The work was supported by the National Natural Science Foundation of China under Grants 61162022 and 61362036, the Natural Science Foundation of Jiangxi China under Grant 20132BAB201021, the Jiangxi Science and Technology Research Development Project of China under Grant KJLD12098, and the Jiangxi Science and Technology Research Project of Education Department of China under Grant GJJ12632.

References

[1] K. P. Pruessmann, "Encoding and reconstruction in parallel MRI," *NMR in Biomedicine*, vol. 19, no. 3, pp. 288–299, 2006.

[2] B. Sharif, J. A. Derbyshire, A. Z. Faranesh, and Y. Bresler, "Patient-adaptive reconstruction and acquisition in dynamic imaging with sensitivity encoding (PARADISE)," *Magnetic Resonance in Medicine*, vol. 64, no. 2, pp. 501–513, 2010.

[3] M. Lustig, D. Donoho, and J. M. Pauly, "Sparse MRI: the application of compressed sensing for rapid MR imaging," *Magnetic Resonance in Medicine*, vol. 58, no. 6, pp. 1182–1195, 2007.

[4] D. Donoho, J. M. Santos, and J. M. Pauly, "Compressed sensing MRI," *IEEE Signal Processing Magazine*, vol. 2, pp. 72–82, 2008.

[5] J. Huang, S. Zhang, and D. Metaxas, "Efficient MR image reconstruction for compressed MR imaging," *Medical Image Analysis*, vol. 15, no. 5, pp. 670–679, 2011.

[6] Z. Zhang, Y. Shi, W. P. Ding, and B. C. Yin, "MR images reconstruction based on TVWL2-L1 model," *Journal of Visual Communication and Image Representation*, vol. 2, pp. 187–195, 2013.

[7] A. Majumdar and R. K. Ward, "On the choice of compressed sensing priors and sparsifying transforms for MR image reconstruction: An experimental study," *Signal Processing: Image Communication*, vol. 27, no. 9, pp. 1035–1048, 2012.

[8] J. Yang, Y. Zhang, and W. Yin, "A fast alternating direction method for TVL1-L2 signal reconstruction from partial Fourier data," *IEEE Journal on Selected Topics in Signal Processing*, vol. 4, no. 2, pp. 288–297, 2010.

[9] S. Ravishankar and Y. Bresler, "MR image reconstruction from highly undersampled k-space data by dictionary learning," *IEEE Transactions on Medical Imaging*, vol. 30, no. 5, pp. 1028–1041, 2011.

[10] E. J. Candes, J. Romberg, and T. Tao, "Robust uncertainty principles: exact signal reconstruction from highly incomplete frequency information," *IEEE Transactions on Information Theory*, vol. 52, no. 2, pp. 489–509, 2006.

[11] D. L. Donoho, "Compressed sensing," *IEEE Transactions on Information Theory*, vol. 52, no. 4, pp. 1289–1306, 2006.

[12] X. Qu, X. Cao, D. Guo, C. Hu, and Z. Chen, "Combined sparsifying transforms for compressed sensing MRI," *Electronics Letters*, vol. 46, no. 2, pp. 121–123, 2010.

[13] F. Knoll, K. Bredies, T. Pock, and R. Stollberger, "Second order total generalized variation (TGV) for MRI," *Magnetic Resonance in Medicine*, vol. 65, no. 2, pp. 480–491, 2011.

[14] D. Donoho and J. Tanner, "Observed universality of phase transitions in high-dimensional geometry, with implications for modern data analysis and signal processing," *Philosophical Transactions of the Royal Society of London A*, vol. 367, no. 1906, pp. 4273–4293, 2009.

[15] Z. B. Xu, X. Y. Chang, and F. M. Xu, "L-1/2 regularization: a thresholding representation theory and a fast solver," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 7, pp. 1013–1027, 2012.

[16] R. Chartrand, "Fast algorithms for nonconvex compressive sensing: MRI reconstruction from very few data," in *Proceedings of the 6th IEEE International Conference on Biomedical Imaging: From Nano to Macro (ISBI '09)*, pp. 262–265, July 2009.

[17] C. Y. Jong, S. Tak, Y. Han, and W. P. Hyun, "Projection reconstruction MR imaging using FOCUSS," *Magnetic Resonance in Medicine*, vol. 57, no. 4, pp. 764–775, 2007.

[18] J. Trzasko and A. Manduca, "Highly undersampled magnetic resonance image reconstruction via homotopic l0-minimization," *IEEE Transactions on Medical Imaging*, vol. 28, no. 1, pp. 106–121, 2009.

[19] C. Chen and J. Z. Huang, "The benefit of tree sparsity in accelerated MRI," *Medical Image Analysis*, vol. 18, pp. 834–842, 2014.

[20] S. Q. Ma, W. T. Yin, Y. Zhang, and A. Chakraborty, "An efficient algorithm for compressed MR imaging using total variation and wavelets," in *Proceeding of the 26th IEEE Conference on*

Computer Vision and Pattern Recognition (CVPR '08, pp. 1–8, Anchorage, Alaska, USA, June 2008.

[21] A. Buades, B. Coll, and J. M. Morel, "A review of image denoising algorithms, with a new one," *Multiscale Modeling & Simulation*, vol. 4, no. 2, pp. 490–530, 2005.

[22] F. F. Dong, H. L. Zhang, and D. X. Kong, "Nonlocal total variation models for multiplicative noise removal using split Bregman iteration," *Mathematical and Computer Modelling*, vol. 55, no. 3-4, pp. 939–954, 2012.

[23] S. Yun and H. Woo, "Linearized proximal alternating minimization algorithm for motion deblurring by nonlocal regularization," *Pattern Recognition*, vol. 44, no. 6, pp. 1312–1326, 2011.

[24] X. Zhang, M. Burger, and X. Bresson, "Bregmanized nonlocal regularization for deconvolution and sparse reconstruction," *SIAM Journal on Imaging Sciences*, vol. 3, no. 3, pp. 253–276, 2011.

[25] W. Dong, X. Yang, and G. Shi, "Compressive sensing via reweighted TV and nonlocal sparsity regularisation," *Electronics Letters*, vol. 49, no. 3, pp. 184–186, 2013.

[26] V. P. Gopi, P. Palanisamy, and K. A. Wahid, "MR image reconstruction based on iterative split Bregman algorithm and nonlocal total variation," *Computational and Mathematical Methods in Medicine*, vol. 2013, Article ID 985819, 16 pages, 2013.

[27] J. Huang and F. Yang, "Compressed magnetic resonance imaging based on wavelet sparsity and nonlocal total variation," in *Proceedings of the 9th IEEE International Symposium on Biomedical Imaging: From Nano to Macro (ISBI '12)*, pp. 968–971, Barcelona, Spain, May 2012.

[28] D. Liang, H. F. Wang, Y. C. Chang, and L. L. Ying, "Sensitivity encoding reconstruction with nonlocal total variation regularization," *Magnetic Resonance in Medicine*, vol. 65, no. 5, pp. 1384–1392, 2011.

[29] H. Mohimani, M. Babaie-Zadeh, and C. Jutten, "A fast approach for overcomplete sparse decomposition based on smoothed L0 norm," *IEEE Transactions on Signal Processing*, vol. 57, pp. 289–301, 2009.

[30] J.-H. Wang, Z.-T. Huang, Y.-Y. Zhou, and F.-H. Wang, "Robust sparse recovery based on approximate L_0 norm," *Acta Electronica Sinica*, vol. 40, no. 6, pp. 1185–1189, 2012.

[31] M. V. Afonso, J. M. Bioucas-Dias, and M. A. T. Figueiredo, "An augmented Lagrangian approach to the constrained optimization formulation of imaging inverse problems," *IEEE Transactions on Image Processing*, vol. 20, no. 3, pp. 681–695, 2011.

[32] P. L. Combettes and V. R. Wajs, "Signal recovery by proximal forward-backward splitting," *Multiscale Modeling & Simulation*, vol. 4, pp. 1168–1200, 2005.

Hindawi Publishing Corporation
Mathematical Problems in Engineering
Volume 2014, Article ID 505372, 22 pages
http://dx.doi.org/10.1155/2014/505372

Research Article

Aerodynamic Optimal Shape Design Based on Body-Fitted Grid Generation

Farzad Mohebbi and Mathieu Sellier

Department of Mechanical Engineering, The University of Canterbury, Private Bag 4800, Christchurch 8140, New Zealand

Correspondence should be addressed to Farzad Mohebbi; farzadmohebbi@yahoo.com

Received 9 April 2014; Accepted 11 June 2014; Published 27 August 2014

Academic Editor: Caner Özdemir

This paper is concerned with an optimal shape design problem in aerodynamics. The inverse problem in question consists in finding the optimal shape an airfoil placed in a potential flow at a given angle of attack should have such that the pressure distribution on its surface matches a desired one. The numerical method to achieve this aim is based on a body-fitted grid generation technique (elliptic, O-type) to generate a mesh over the airfoil surface and solve for the flow equation. The O-type scheme is used due to its ability to generate a high quality (fine and orthogonal) grid around the airfoil surface. This paper describes a novel and very efficient sensitivity analysis scheme to compute the sensitivity of the pressure distribution to variation of grid node positions and both the conjugate gradient method (CGM) and a version of the quasi-Newton method (i.e., BFGS) are used as optimization algorithms to minimize the difference between the computed pressure distribution on the airfoil surface and desired one. The elliptic grid generation technique allows us to map the physical domain (body) onto a fixed computational domain and to discretize the flow equation using the finite difference method (FDM).

1. Introduction

Thanks to the advent of modern high speed computers over the last few decades, computational fluid dynamics (CFD) has been extensively employed as an analysis and as a design optimization tool. Among the methodologies often employed in shape optimization are gradient-based techniques. These techniques may be applied to minimize a specified objective function. In airfoil shape optimization, the objective function can be, for example, a measure of difference between the pressure distribution on the airfoil surface and a desired one, and it would be desirable to minimize this objective function. In this paper, we consider the 2D shape optimization of an airfoil in an irrotational and incompressible flow governed by the Laplace equation. The procedure employed is based on the elliptic grid generation, a novel sensitivity analysis (based on finite difference method), and an optimization method. The conjugate gradient method and an efficient version of quasi-Newton method (BFGS) will be used as the optimization algorithms. The airfoil surface is parameterized using the grid points and the Bezier curve. Three different types of design variables were considered: the grid points, the Bezier curve control points, and the maximum thickness of NACA00xx airfoils. It will be represented that the use of the Bezier curve significantly improves the optimization performance to reach the optimal shape. Furthermore, it will be shown that the proposed sensitivity analysis method reduces the computation cost significantly even for large number of the design variables.

Some of the earliest studies using a combination of CFD with numerical optimization in aerodynamic were made by Hicks et al. [1] and Hicks and Henne [2]. In [1], a procedure for optimal design of symmetric low-drag, nonlifting transonic airfoils in inviscid flow is proposed. The proposed procedure uses an optimization program based on the method of the feasible directions coupled with an analysis program that utilizes a relaxation method to solve the partial differential equation that governs the inviscid, transonic, and small disturbance fluid flow. The drag minimization with geometric constraints is considered in this reference. In fluid dynamics, Pironneau was the first one to use the adjoint equations for design [3]. This is the first application of control

theory to design optimization. However, within the field of aeronautical computational fluid dynamics, Jameson was the first researcher who used the continuous adjoint formulation for aerodynamic shape optimization in transonic potential flows and flows governed by Euler equations [4–7]. Giles et al. made considerable contributions to the development of the discrete adjoint approach [8–11]. In [10], the adjoint equations are formulated for the transonic design applications for which there are shocks. The adjoint equations were already formulated for the incompressible or subsonic flows in which the assumption that the original nonlinear flow solution is smooth is valid. In [11], a number of algorithm developments are presented for adjoint methods using the "discrete" approach. In continuous adjoint method, the original partial differential equations are linearized, the adjoint partial differential equation and appropriate boundary conditions are formulated, and finally the equations are discretized. Unlike the continuous adjoint approach, in discrete adjoint approach the partial differential equations are discretized, the discrete equations are linearized, and then the transpose of the linear operator is used to form the adjoint problem. The adjoint equations have also been used by Baysal and Eleshaky to infer the optimal design for a scramjet-afterbody configuration which yields the maximum axial thrust [12] and by Ta'asan et al. to obtain an optimal airfoil shape [13]. Baysal and Eleshaky's work was based on a computational fluid dynamics-sensitivity analysis algorithm (two different quasi-analytical approaches: the direct method and the adjoint variable method) to solve Euler equations for the inviscid analysis of the flow. Adjoint methods have been applied to incompressible viscous flow problems by Cabuk et al. [14] and Desai and Ito [15]. Cabuk et al. worked on the problem of determining the profile of a channel or duct that provides the maximum static pressure rise by solving the incompressible, laminar flow governed by the steady state Navier-Stokes equations. Early applications of discrete adjoint methods on unstructured meshes can be found in works by Elliott and Peraire in inviscid [16] and viscous flows [17] for 2D and 3D [18] configurations. In [16], an inverse design procedure for single- and multielement airfoils using unstructured grids and based on the Euler equations is presented. The discrete adjoint method is used to compute the sensitivities and the results are compared with corresponding finite difference values. It is shown that the use of the adjoint method practically eliminates the dependence of the objective function gradient computation on the number of design variables. The continuous adjoint approach for unstructured grids has been developed by Anderson and Venkatakrishnan [19]. In [19], aerodynamic shape optimization on unstructured grids using a continuous adjoint approach is developed and analyzed for inviscid and viscous flows. B spline and Bezier curves are employed to parameterize the airfoil surface. The objective functions considered include drag minimization, lift maximization, and matching a specified pressure distribution. The quasi-Newton optimization method is used to obtain the optimal design. Evolutionary algorithms, as methods that do not need the computation of the gradient, have recently gained much attention in the context of aerodynamic shape optimization [20–24]. Although they are of extremely high computational cost, they have the advantage that they can escape from a "local minimum" (a major issue in using gradient based methods) and have the ability of finding globally optimum solutions amongst many local optima [25, 26]. A detailed study of many methods in shape optimization in fluid mechanics is given by Mohammadi and Pironneau [27].

The adjoint approach, as an alternative to the finite difference method to compute the gradient of functional with respect to the design variables, is computationally very efficient. Therefore, as far as the computational cost is concerned, it is the appropriate choice. This is the case when there are a large number of design variables which makes use of the finite difference method impractical. The differences between the adjoint method and the finite difference one (to compute the gradient of functional with respect to the design variables) can be summarized as follows.

Adjoint Method. N design variables, 1 flow solution, and 1 adjoint calculation.

Finite Difference Method. N design variables, N flow solutions. Because

$$\frac{\partial \mathcal{J}}{\partial \alpha_j} = \frac{\left[\mathcal{J}\left(\alpha_j + \delta\alpha_j\right) - \mathcal{J}\left(\alpha_j\right)\right]}{\delta\alpha_j}, \tag{1}$$

where $\mathcal{J}$ is the objective function and α_j are the design variables [28]. As can be seen, aerodynamic shape optimization with large number of design variables is computationally practical only when the adjoint method is used. However, as will be shown in this paper, a novel sensitivity analysis will be presented which makes use of the finite difference method comparable (from computation cost viewpoint) to the use of adjoint method. The numerical algorithm used in this paper is already employed in shape optimization problems in heat conduction [29, 30]. The numerical algorithm consists of three steps, namely, grid generation and flow equation solver to find the pressure on the airfoil surface, sensitivity analysis to compute the gradient of the objective function with respect to the design variables, and an optimization method to minimize the functional and reach optimum solution.

2. Governing Equation

For a two-dimensional incompressible flow, a stream function ψ can be defined such that

$$u = \frac{\partial \psi}{\partial y}, \qquad v = -\frac{\partial \psi}{\partial x}, \tag{2}$$

where u and v are the components of the velocity vector $\mathbf{V}$; that is, $\mathbf{V} = u\mathbf{i} + v\mathbf{j}$ ($\mathbf{i}$ and $\mathbf{j}$ are the unit vectors in x and y directions, resp.). Combining the above definitions with the irrotationality condition leads to the following Laplace equation for the stream function

$$\frac{\partial^2 \psi}{\partial x^2} + \frac{\partial^2 \psi}{\partial y^2} = 0. \tag{3}$$

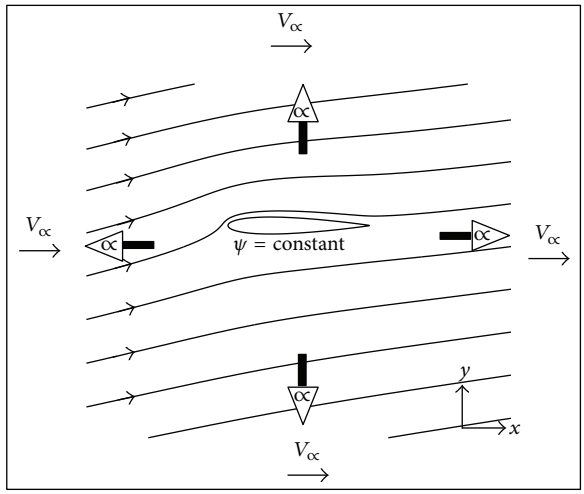


Figure 1: Boundary conditions at infinity and on the airfoil surface (no-penetration).

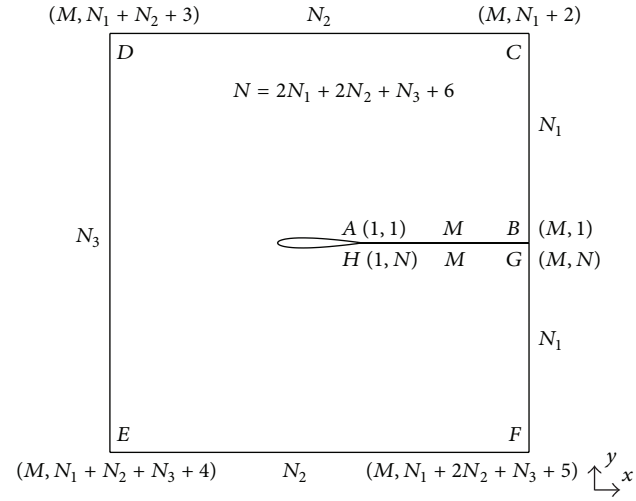


Figure 2: Physical domain showing the discretization of the boundaries used for O-type elliptic grid generation technique.

Consider an irrotational incompressible flow over an airfoil (Figure 1). The boundary conditions are as shown in Figure 1.

Conditions at Infinity. Far away from the airfoil surface (toward infinity), in all directions, the flow approaches the uniform freestream conditions. If the angle of attack (AOA) is α, the free stream velocity V_∞, the components of the flow velocity can be written as

$$u = \frac{\partial \psi}{\partial y} = V_\infty \cos\alpha,$$
$$v = -\frac{\partial \psi}{\partial x} = V_\infty \sin\alpha. \tag{4}$$

Condition on the Airfoil Surface. The relevant boundary condition at the airfoil surface for this inviscid flow is the no-penetration boundary condition. Thus the velocity vector must be *tangent* to the surface. This wall boundary condition can be expressed by

$$\frac{\partial \psi}{\partial s} = 0, \quad \text{or } \psi = \text{constant}, \tag{5}$$

where s is tangent to the surface. In the problem of the flow over an airfoil, if the free stream velocity and the angle of attack are known, from the boundary conditions at infinity (see (4)) and the wall boundary condition (see (5)) one can compute the stream function ψ at any point of the physical domain (flow region). Then, by knowing ψ, one can compute the velocity of all points. Since for an incompressible flow, the pressure coefficient is a function of the velocity only, one can obtain the pressure of any point in the flow region, as will be shown.

Pressure Coefficient. The pressure coefficient C_p is defined as

$$C_p = \frac{p - p_\infty}{(1/2)\,\rho_\infty V_\infty^2} = 1 - \left(\frac{V}{V_\infty}\right)^2, \tag{6}$$

where V is the velocity of fluid at the point at which the pressure coefficient C_p is being evaluated. At standard sea level conditions,

$$\rho_\infty = 1.23\ \text{kg/m}^3,$$
$$p_\infty = 1.01 \times 10^5\ \text{N/m}^2, \tag{7}$$

where ρ_∞ and p_∞ are the freestream density and pressure, respectively. From (6),

$C_p = 0$ indicates that the point at which the pressure coefficient C_p is being evaluated is located at infinity.

$C_p = 1$ indicates that the point at which the pressure coefficient C_p is being evaluated is a stagnation point (where $V = 0$). For an incompressible flow, this is the maximum allowable value of C_p anywhere in the flow field.

And in regions of the flow where $V > V_\infty$, C_p value will be negative.

3. Grid Generation and Flow Solver

To calculate the pressure at any point in the flow region, a grid should be generated over the region. The grid generation method considered in this study is the elliptic grid generation, which was proposed by Thompson et al. [31] and is based on solving a system of elliptic partial differential equations to distribute nodes in the interior of the physical domain by mapping the irregular physical domain from the x and y physical plane (Figure 2) onto the ξ and η computational plane (Figure 3), which is a regular region.

The O-type elliptic grid generation technique is employed here which results in a smooth and orthogonal grid over the airfoil surface. The discretization of the physical domain (flow region) and the corresponding computational domain are shown in Figures 2 and 3, respectively. In the computational domain, M and $N = 2N_1 + 2N_2 + N_3 + 6$ are the number of nodes in the ξ and η directions, respectively. The resulting O-type gird scheme over an airfoil for the case $N_2 = N_1$ and $N_3 = 2N_1 - 1$ or $N = 6N_1 + 5$ is shown in Figure 4.

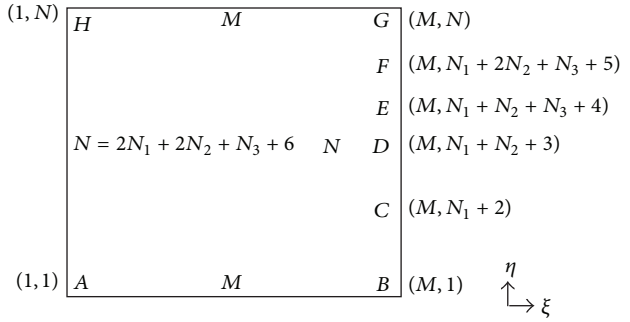


Figure 3: Computational domain showing the discretization of the physical domain boundaries.

The initial guess for the elliptic grid generation is performed using the transfinite interpolation (TFI) method. Since TFI method is an algebraic technique and does not require much computational time, it will be an appropriate initial guess for the elliptic grid generation method and accelerates convergence time for the elliptic method. Another advantage of using the TFI method as an initial guess is that it prevents the grids generated by the elliptic (O-type) method from folding.

If V_∞ and α are known, then from (4) one can obtain the stream function ψ at any point on the boundaries of the physical domain as follows:

$$\psi_2 = \psi_1 + (y_2 - y_1) V_\infty \cos\alpha, \tag{8}$$

$$\psi_2 = \psi_1 - (x_2 - x_1) V_\infty \sin\alpha, \tag{9}$$

where subscripts 1 and 2 refer to any two arbitrary grid points on the boundaries of the physical domain. Equations (8) and (9) are applied to vertical and horizontal boundaries of the physical domain, respectively. By knowing the values of the stream function ψ on the boundaries of the physical domain as well as on the airfoil surface, we can obtain the values of ψ over the physical domain by applying the Kutta condition [32, 33] and using the following formula (by mapping the physical domain onto the computational domain [29]):

$$\begin{aligned} &\alpha\psi_{\xi\xi} - 2\beta\psi_{\xi\eta} + \gamma\psi_{\eta\eta} \\ &\quad = -J^2\left(P(\xi,\eta)\psi_\xi + Q(\xi,\eta)\psi_\eta\right), \end{aligned} \tag{10}$$

where

$$\begin{aligned} \alpha &= x_\eta^2 + y_\eta^2, \\ \beta &= x_\xi x_\eta + y_\xi y_\eta, \\ \gamma &= x_\xi^2 + y_\xi^2, \\ J &= x_\xi y_\eta - x_\eta y_\xi, \quad \text{(Jacobian of transformation)}. \end{aligned} \tag{11}$$

P and Q are grid control functions which control the density of grids towards a specified coordinate line or about a specific grid point. Equations (10) and (11) are discretized using the finite difference method. For more details, please refer to [29].

Velocity Calculation. There are three sections where the velocity must be known:

(1) the outer boundaries (four sides CD, DE, EF, and FC of the rectangle shown in Figure 2);

(2) the airfoil surface (AH in Figure 2);

(3) the inside of the physical domain.

The velocity values on the outer boundaries are known from the conditions at infinity (using (4)). In other words, x-component of the velocity vector (u) on all the outer boundaries is equal to $V_\infty \cos\alpha$ and y-component of the velocity vector (v) on all the outer boundaries is equal to $V_\infty \sin\alpha$. For the inside of the physical domain and the airfoil surface, we can use the flowing relationships to evaluate the velocity. These relationships are obtained by using the transformation relationships and chain rule in mapping the physical domain onto the computational one. Consider

$$u_{i,j} = \left.\frac{\partial\psi}{\partial y}\right|_{i,j} = \frac{1}{J}\left[-\left(x_\eta\right)_{i,j}\left(\psi_\xi\right)_{i,j} + \left(x_\xi\right)_{i,j}\left(\psi_\eta\right)_{i,j}\right], \tag{12}$$

$$v_{i,j} = -\left.\frac{\partial\psi}{\partial x}\right|_{i,j} = -\frac{1}{J}\left[\left(y_\eta\right)_{i,j}\left(\psi_\xi\right)_{i,j} - \left(y_\xi\right)_{i,j}\left(\psi_\eta\right)_{i,j}\right]. \tag{13}$$

The central and forward difference schemes are used for the inside of the physical domain and the airfoil surface, respectively. After obtaining the components of the velocity vector, the total velocity (velocity distribution) can be computed by

$$V_{i,j} = \sqrt{u_{i,j}^2 + v_{i,j}^2}. \tag{14}$$

As stated before, for an incompressible flow, the pressure coefficient can be expressed in terms of the velocity only. Thus (6) can be used to determine the pressure at any grid point in the domain. Therefore,

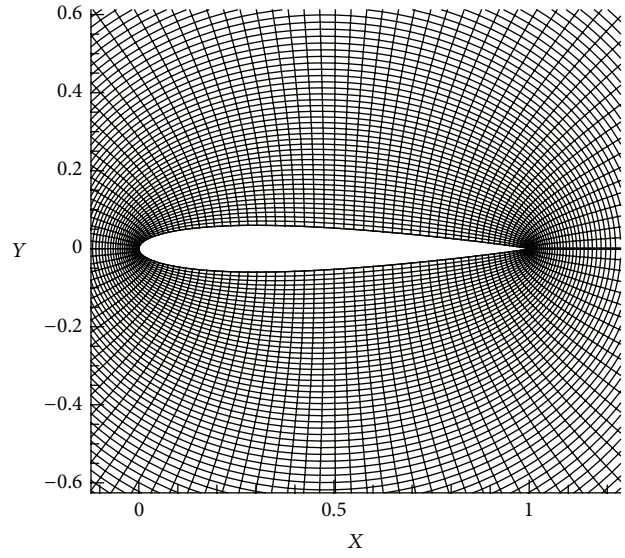
$$p_{i,j} = \frac{1}{2}\rho\left(V_\infty^2 - V_{i,j}^2\right) + p_\infty. \tag{15}$$

Validation of the Results for the Pressure Distribution. The results obtained here are compared with the results given in [34] which are obtained both analytically and by using the panel method (see Figure 7).

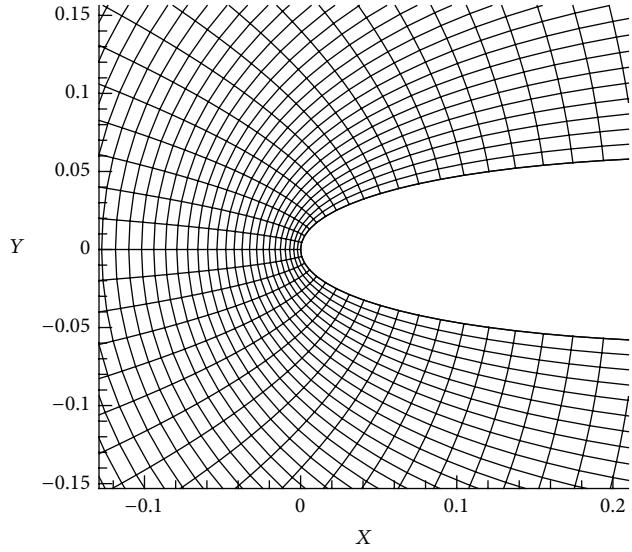
Validation Case. The pressure coefficient distribution (C_p) over the NACA 0012 airfoil at an angle of attack $\alpha = 9^\circ$ is plotted. The results are compared with the results from [34]. The O-type grid size used in the computation is 155×155. The computation time is 53 seconds.

4. Airfoil Parameterization

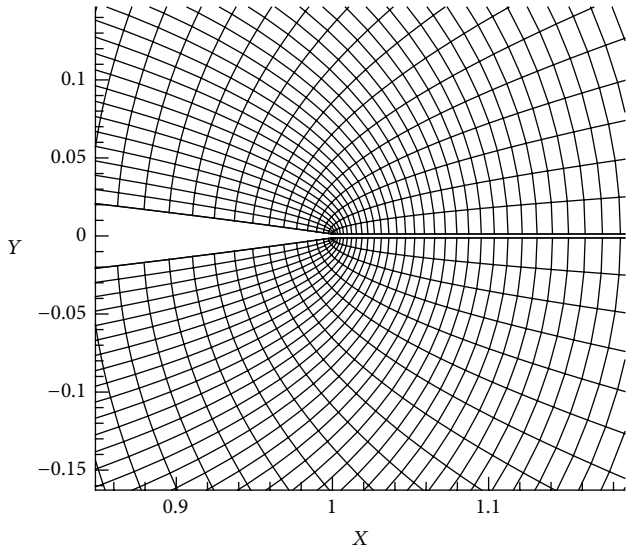
So far, the airfoil surface is parameterized by grid points which result in accurate pressure distribution on the airfoil



Close-up view of O-type grid around the airfoil



Magnifed view of grid around the leading edge



Magnifed view of grid around the trailing edge

FIGURE 4: O-type grid (elliptic) around an airfoil. This close-up view of the grid shows orthogonality and smoothness of the gridlines especially near airfoil surface.

surface (see Figures 14, 19, and 25). However, a large number of grid points are needed to obtain such accurate results which in turn lead to high (see Figures 5 and 6) computation cost. The design variables are the coordinate (usually y-coordinate) of grid points. Therefore, the optimization process may be inappropriate if there are a large number of design variables since it is difficult to maintain a smooth geometry, the optimization problem will be difficult to solve, and the optimization strategy is likely to fail or be impractical [35]. Thus alternative methods of airfoil surface parameterization are needed. These methods should represent great flexibility in defining the airfoil surface with minimum design variables. In this paper, in addition to the grid points to represent the airfoil surface, Bezier curves (a special subset of B-spline) are employed due to their ability to produce airfoil surfaces easily and precisely with only a few control points.

Bezier Curve. A Bezier curve is a special case of a B-spline curve and is mathematically defined by

$$P(t) = \sum_{i=0}^{n} B_i J_{n,i}(t), \tag{16}$$

where

$$J_{n,i}(t) = \frac{n!}{i!\,(n-i)!} t^i (1-t)^{n-i} \tag{17}$$

is Bernstein basis polynomial of degree n. By convention $0^0 \equiv 1$ and $0! \equiv 1$. Here, n, the degree of the Bernstein

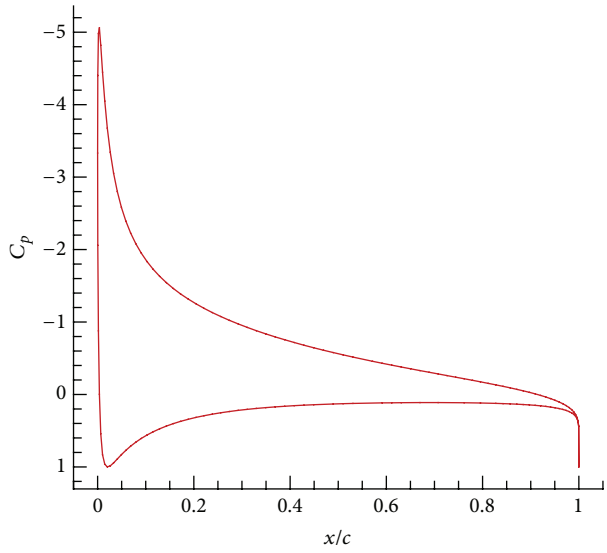


Figure 5: The pressure coefficient distribution over an NACA 0012 airfoil at a 9° angle of attack obtained numerically.

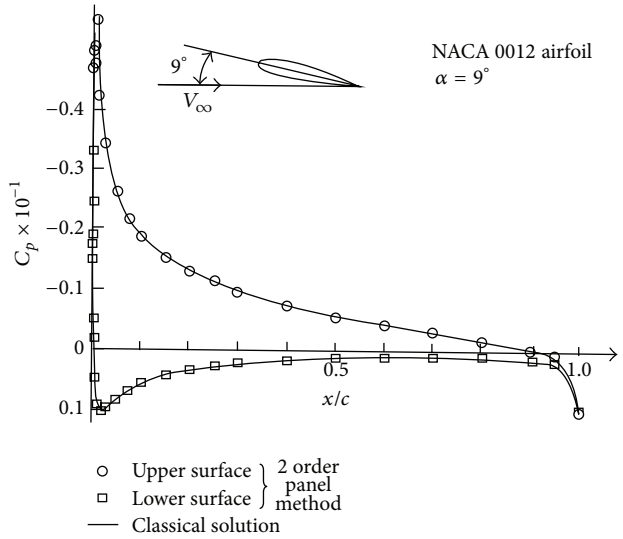


Figure 6: The pressure coefficient distribution over an NACA 0012 airfoil at a 9° angle of attack [34].

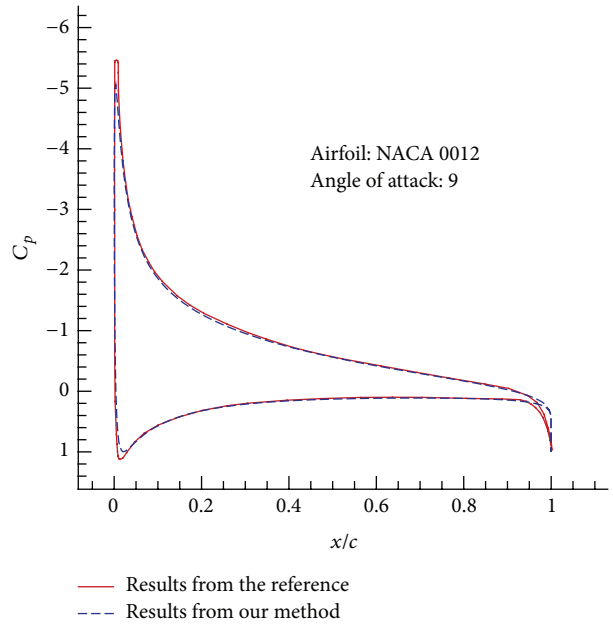


Figure 7: Comparison between the results from [34] and the results from our method for validation case. The figure shows an excellent agreement between the results.

basis polynomial is one less than the number of points in the Bezier polygon. In other words, the number of control points is $n + 1$. The points B_i are the vertices of a Bezier polygon or the control points of a Bezier curve. The curve begins at B_0 and ends at B_n. The order of a Bezier curve k is equal to $n + 1$. In other words, the order of a Bezier curve is equal to the number of the control points [36].

In this paper, two different Bezier curves of order 7 (degree = 6) and of order 11 (degree = 10) will be considered. As it will be shown, the Bezier curve of order 7 represents the better optimization performance due to its less design variables. However, this kind of Bezier curve is not able to produce very accurate airfoil shapes. Indeed, it is appropriate to NACA 00xx airfoils only. On the other hand, the Bezier curve of order 11 can successfully generate any airfoil shape with a high degree of accuracy. Therefore, the formulation for the Bezier curve of order 11 only will be given here. The formulation for the Bezier curve of order 7 can be written in a similar fashion.

The parametric Bezier curve of order 11 is as follows:

$$\begin{aligned} n = 10 &\Longrightarrow \text{number of control points} = 11 \\ P(t) &= \sum_{i=0}^{10} B_i J_{10,i}(t) \\ &= B_0 J_{10,0}(t) + \cdots + B_{10} J_{10,10}(t) \\ &= B_0 \frac{10!}{0!\,(10-0)!} t^0 (1-t)^{10-0} \\ &\quad + \cdots + B_{10} \frac{10!}{10!\,(10-10)!} t^{10} (1-t)^{10-10}. \end{aligned} \tag{18}$$

Therefore,

$$\begin{aligned} P(t) = {} & B_0 (1-t)^{10} + B_1 10t (1-t)^9 + B_2 45t^2 (1-t)^8 \\ & + B_3 120t^3 (1-t)^7 + B_4 210t^4 (1-t)^6 \\ & + B_5 252t^5 (1-t)^5 + B_6 210t^6 (1-t)^4 \\ & + B_7 120t^7 (1-t)^3 + B_8 45t^8 (1-t)^2 \\ & + B_9 10t^9 (1-t)^1 + B_{10} t^{10}. \end{aligned} \tag{19}$$

In order to construct the airfoil surface, two Bezier curve will be considered corresponding to the upper and lower surfaces,

respectively. Here there are 11 control points (vertices) for each surface. Since the coordinates of the airfoil surface are known, the problem is to determine values for the control points B_i $(i = 0, \ldots, 10)$. In other words, our problem is to specify the coordinates of the control points B_i so that the curve passes through the predetermined data points on the airfoil surface. Equation (16) can be written in matrix form as follows:

$$[P(t)] = [J(t)][B]. \tag{20}$$

If the number of the chosen data points on the airfoil surface is m and the degree of Bezier curve is n, then $[P(t)]$ is a $m \times 2$ matrix, $[J(t)]$ is a $m \times (n+1)$ matrix, and $[B]$ is a $(n+1) \times 2$ matrix. Two columns of the matrix $[P(t)]$ pertain to the x- and y-coordinates of the predetermined data on the airfoil surface. Equation (20) can be rewritten as

$$[P(t)]_{m\times 2} = [J(t)]_{m\times(n+1)}[B]_{(n+1)\times 2}. \tag{21}$$

If $m = n + 1$, the matrix $[J(t)]_{m\times(n+1)}$ will be a square matrix and it can be inverted. In such a case, (21) can be written as follows to find the matrix $[B]$:

$$[B]_{(n+1)\times 2} = [J(t)]^{-1}_{m\times(n+1)}[P(t)]_{m\times 2}. \tag{22}$$

However, the number of the airfoil surface data points is usually more than the number of control points. In such a case, there are more equations than unknowns and the matrix $[J(t)]_{m\times(n+1)}$ is no longer a square matrix. Hence it is required to convert it to a square matrix by multiplying both sides of (21) by the transpose of $[J(t)]_{m\times(n+1)}$ as follows:

$$\begin{aligned}&[J(t)]^T_{(n+1)\times m}[P(t)]_{m\times 2}\\ &\quad = [J(t)]^T_{(n+1)\times m}[J(t)]_{m\times(n+1)}[B]_{(n+1)\times 2}.\end{aligned} \tag{23}$$

Thus,

$$\begin{aligned}[B]_{(n+1)\times 2} = &\left[[J(t)]^T_{(n+1)\times m}[J(t)]_{m\times(n+1)}\right]^{-1}\\ &\cdot [J(t)]^T_{(n+1)\times m}[P(t)]_{m\times 2}.\end{aligned} \tag{24}$$

NACA 0015 and TsAGI "B" 12% airfoils produced by Bezier curve with $n = 10$ and $m = 51$ and their comparison with conventional NACA 0015 and TsAGI "B" 12% airfoils are shown in Figures 8 and 9, respectively. There is an excellent agreement between two airfoils in each figure.

The predetermined data for the NACA airfoils can be extracted from, for example, the software JavaFoil [37] which is based on the analytical NACA formulations.

NACA 00xx Symmetric Airfoils. Since the maximum thickness of a NACA 00xx symmetric airfoil will be considered as a design variable, the equation for generating such airfoils is given as follows:

$$\begin{aligned}\pm y_t = \frac{t}{0.2}\Big[&0.2969\sqrt{x} - 0.1260x\\ &-0.3516x^2 + 0.2843x^3 - 0.1015x^4\Big],\end{aligned} \tag{25}$$

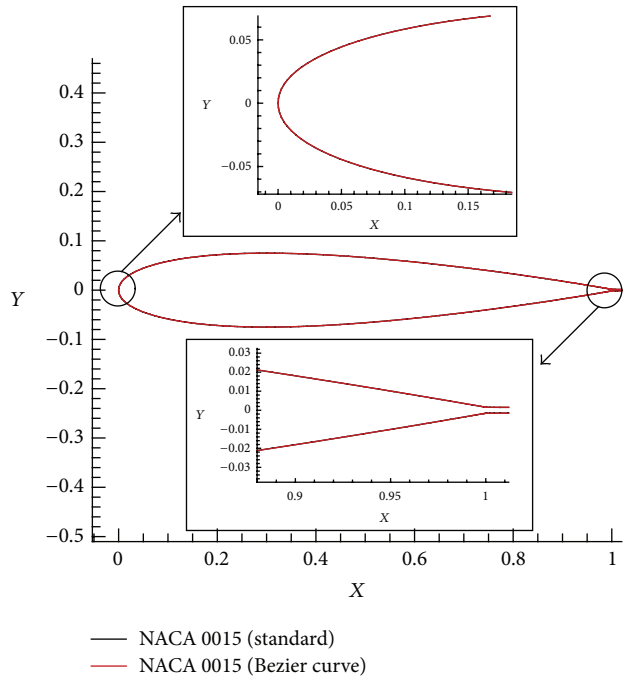


Figure 8: Comparison between the standard airfoil and the Bezier curve for a NACA0015.

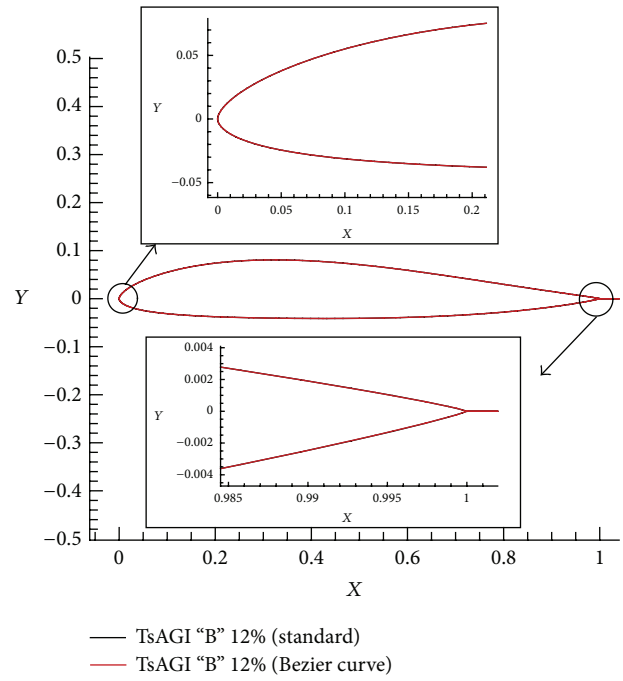


Figure 9: Comparison between the standard airfoil and the Bezier curve for a TsAGI "B" 12%.

where x is coordinates along the chord of the airfoil, from 0 to c (c is the *chord length* and is assumed equal to 1), y_t is the thickness coordinates above and below the line extending along the length of the airfoil, and t is maximum thickness of the airfoil in percentage of chord (i.e., t in a %15 thick airfoil would be 0.15). Equation (25) can be used to find

the y-coordinates of a NACA 00xx symmetric airfoil by knowing the values for x and t. As will be shown, the maximum thickness of such airfoils will also be considered as a design variable. By optimizing the thickness, the optimal shape for such airfoils will be obtained. This kind of optimization problem, however, is not comprehensive and produces the optimal NACA 00xx symmetric airfoils only. In summary, three kinds of design variable will be considered in this paper for airfoil shape optimization which are grid points on a given airfoil surface extracted from, say, the software JavaFoil, the Bezier curve control points, and the maximum thickness of NACA 00xx symmetric airfoils.

5. Shape Optimization

Different objective functions may be considered for the aerodynamics shape optimization including maximizing the lift-drag ratio, maximizing the lift, and minimizing the drag. In the framework of this paper, the shape optimization problem will be to infer the shape an airfoil should have so that the pressure distribution on the airfoil surface matches a prescribed one (an inverse problem). In inverse design problem, the desired pressure distribution of the target design may be specified a priori.

Design Variable (DV). Here the airfoil grid points, the Bezier curve control points, and the maximum thickness of NACA00xx airfoils are considered as design variables. Therefore, one has the following:

Case 1: the airfoil grid points as design variable (see Figure 13);

Case 2: the Bezier curve control points as design variable;

Case 3: the maximum thickness of NACA00xx airfoils.

Case 1. The mathematical expression for the objective function considered for Case 1 can be stated as

$$\mathcal{J} = \sum_{j=2, j\neq (N+1)/2}^{N-1} \left(P_{1,j} - P_{d(1,j)}\right)^2, \tag{26}$$

where $P_{(1,j)}$ is the pressure at grid points $F_{1,j}$ on the airfoil surface and $P_{d(1,j)}$ is the desirable pressure at grid points $F_{1,j}$ on the airfoil surface (Figure 10). The aim is to minimize $\mathcal{J}$ and to reach the desirable pressure distribution by changing the position of the grid points on the airfoil surface. Since the x-coordinates of the grid points can be constant during the optimization process, only the y-coordinates of the grid points are considered as design variables. Two end points of airfoil, namely, *leading edge* ($j = (N+1)/2$) and *trailing edge* ($j = 1, N$), are fixed. Thus they are not considered as design variables.

Case 2. The mathematical expression for the objective function considered for Case 2 can be stated as

$$\mathcal{J} = \sum_{i=1}^{2m-4} \left(P_{iB} - P_{iB_d}\right)^2, \tag{27}$$

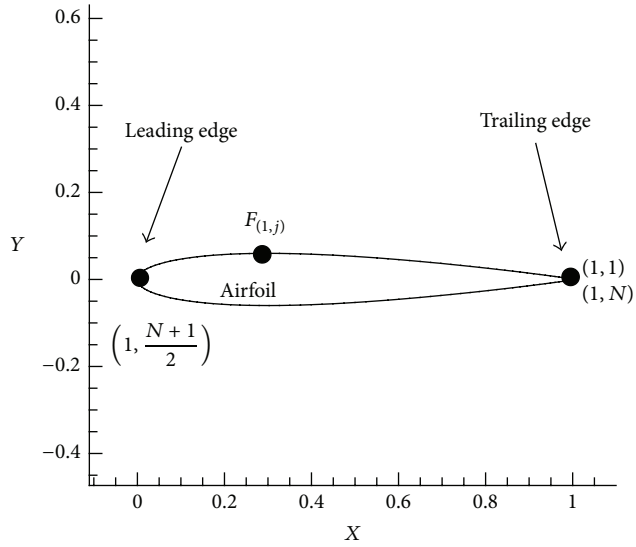


Figure 10: Illustration of the airfoil surface points to be optimized so that the objective function reaches a minimum.

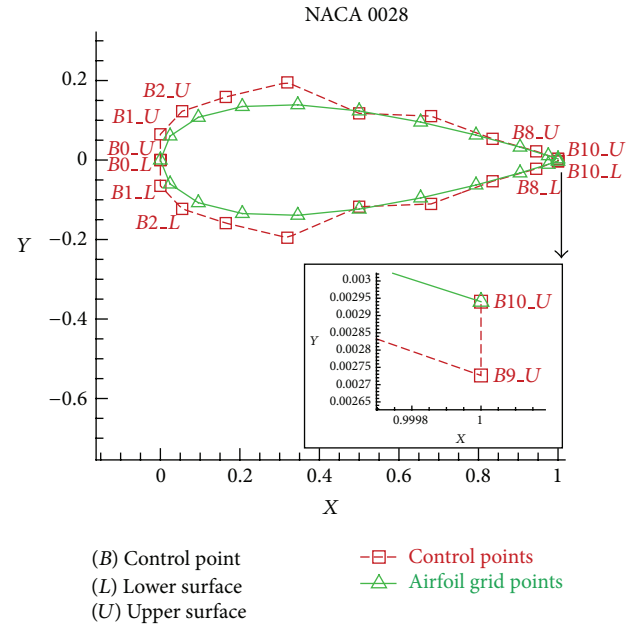


Figure 11: Illustration of the Bezier control points (B_i) to be optimized so that the objective function (see (27)) reaches a minimum.

where m is the number of the predetermined data on each of the upper and the lower surfaces of the airfoil, P_{iB} is the pressure at point i of the airfoil surface generated by the Bezier curve, and P_{iB_d} is the desirable pressure at point i. Why does $2m-4$? m data points for the upper surface, m data points for the lower surface, and the leading and the trailing edges for two surfaces are considered fixed. The aim is to minimize $\mathcal{J}$ and to reach the desirable pressure distribution by changing the y-position of the control points B_i ($i = 1, \ldots, 9$) on each of the upper and the lower surfaces of the airfoil (see Figure 11). B_0 and B_{10}, which are concerned with the leading

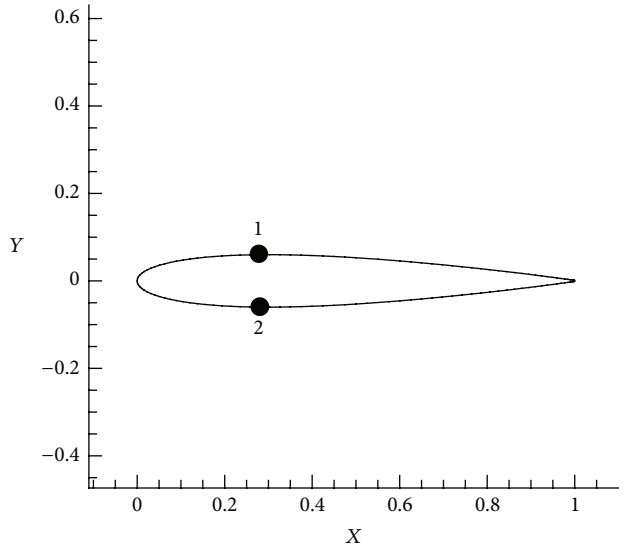


Figure 12: The location for the maximum thickness on the upper and lower airfoil surfaces. y-coordinates of the points 1 and 2 are considered as the design variables.

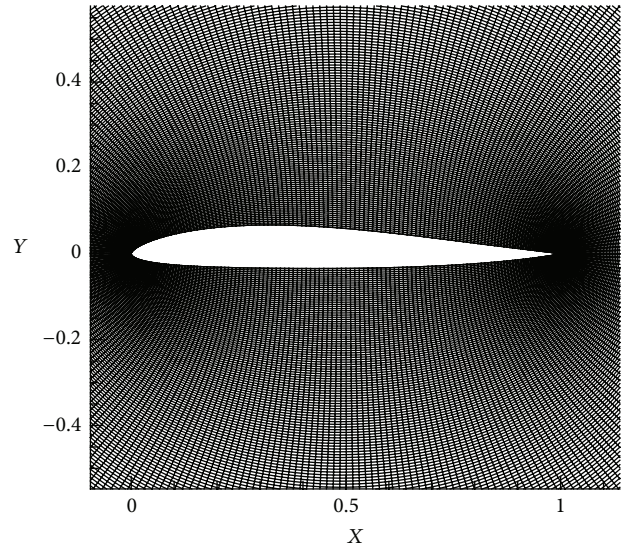


Figure 13: Grid used in Test Case 1 (around initial shape).

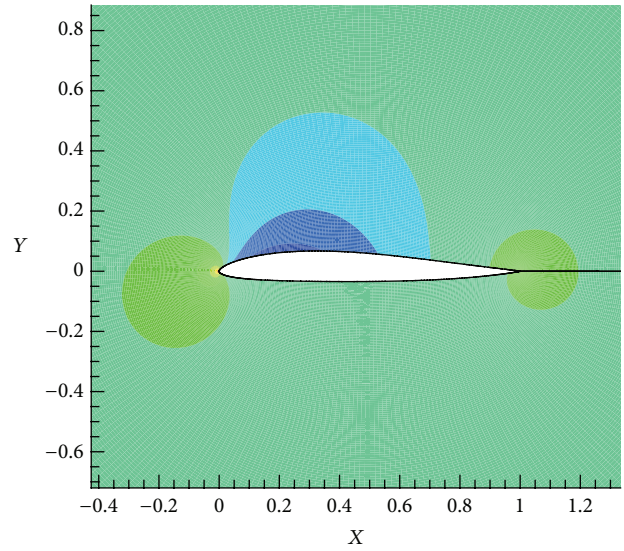


Figure 14: Pressure distribution around the airfoil surface (initial shape).

edge and the trailing edge, respectively, are considered fixed for both upper and lower surfaces. Therefore, for the shape optimization problem with a Bezier curve of order 11, we have $2 \times (11 - 2) = 18$ design variables. For the shape optimization problem with a Bezier curve of order 7, we have $2 \times (7 - 2) = 10$ design variables. The reason for considering these two kinds of the Bezier curve is twofold:

(1) to show that the optimization problem will be more successful if we have less number of design variable;

(2) to have a very accurate and flexible representation of the airfoil shapes, a degree of at least 10 should be used.

Case 3. The airfoil surface is generated by the analytical NACA formula (25) and the maximum thickness is considered as the design variable. To show the accuracy of the sensitivity scheme, the upper and lower airfoil surfaces are generated separately and hence the design variables will be two maximum thicknesses in the upper and lower airfoil surfaces. As shown in Figure 12, if the indices 1 and 2 denote the location of maximum thickness on the upper and lower airfoil surfaces, respectively, then the mathematical expression for the objective function considered for Case 3 is as follows:

$$\mathcal{J} = \sum_{i=1}^{2} \left(P_i - P_{d(i)}\right)^2. \tag{28}$$

6. Sensitivity Analysis

Suppose we wish to calculate the sensitivity of pressure of nodes on the airfoil surface (see Figure 10), $P_{1,j}$ $(j = 2, \ldots, N-1, j \neq (N+1)/2)$, to the y-position of the nodes on the airfoil surface, $y_{1,j'}$ $(j' = 2, \ldots, N-1, j' \neq (N+1)/2)$.

The sensitivity analysis can be performed by introducing small perturbations to the y-coordinate of each point on the airfoil surface, individually. The grid generation and flow problem may be solved for this perturbed shape to obtain the new values for the pressure $P_{1,j}$. Using these values for the pressure, the dependency of the pressure $P_{1,j}$ to the perturbation of the y-position of points of coordinates $(1, j')$, $y_{1,j'}$, can be evaluated. The finite difference method may be used to formulate these sensitivities as follows:

$$\frac{\partial P_{1,j}}{\partial y_{1,j'}} = \frac{P_{1,j}\left(y_{1,j'} + \varepsilon y_{1,j'}\right) - P_{1,j}\left(y_{1,j'}\right)}{\varepsilon y_{1,j'}}, \tag{29}$$

where ε may be, say, 10^{-6}. The term $\varepsilon y_{1,j'}$ is the perturbation in the y-position of points of coordinates $(1, j')$, $y_{1,j'}$.

Since the sensitivity of each pressure $P_{1,j}$ ($j = 2,\ldots,N-1, j \neq (N+1)/2$) to each y-position of points of coordinates $(1, j')$ ($j' = 2,\ldots,N-1, j' \neq (N+1)/2$) is required, the computation of the sensitivity coefficients using this method requires $(N-3)$ additional solutions of the flow problem. Therefore, this method is only suitable when the number of points on the airfoil surface is small. Thus for the airfoil shape optimization problem, which demand a fine grid to obtain accurate results, the perturbation method using the finite difference method will be of high computation cost. In this paper, we will expand the novel method used in evaluating the sensitivity matrix in the shape optimization of heat transfer problems. As will be shown, it requires only one solution of the flow problem (at each iteration) to compute all sensitivity coefficients.

With regard to (26), the following equation can be written in order to calculate the Jacobian matrix

$$\frac{\partial \mathcal{J}}{\partial y_{1,l}} = 2 \sum_{j=2, j\neq (N+1)/2}^{N-1} \left(P_{1,j} - P_{d(1,j)}\right) \frac{\partial P_{1,j}}{\partial y_{1,l}}, \tag{30}$$

where $(j = 2,\ldots,N-1, j \neq (N+1)/2)$ and $(l = 2,\ldots,N-1, l \neq (N+1)/2)$. The expression $\partial P_{1,j}/\partial y_{1,l}$ in the above relation is called the Jacobian coefficient. In this case, the sensitivity matrix can be expanded as

$$\mathbf{Ja}_y = \begin{bmatrix} \frac{\partial P_{1,2}}{\partial y_{1,2}} & \frac{\partial P_{1,2}}{\partial y_{1,3}} & \frac{\partial P_{1,2}}{\partial y_{1,4}} & \cdots & \frac{\partial P_{1,2}}{\partial y_{1,N-1}} \\ \frac{\partial P_{1,3}}{\partial y_{1,2}} & \frac{\partial P_{1,3}}{\partial y_{1,3}} & \frac{\partial P_{1,3}}{\partial y_{1,4}} & \cdots & \frac{\partial P_{1,3}}{\partial y_{1,N-1}} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \frac{\partial P_{1,N-1}}{\partial y_{1,2}} & \frac{\partial P_{1,N-1}}{\partial y_{1,3}} & \frac{\partial P_{1,N-1}}{\partial y_{1,4}} & \cdots & \frac{\partial P_{1,N-1}}{\partial y_{1,N-1}} \end{bmatrix}. \tag{31}$$

Since the physical domain is mapped onto the computational one, the chain rule may be used to correlate variables in the two domains. Therefore,

$$\frac{\partial P_{1,j}}{\partial x_{1,l}} = \frac{\partial P_{1,j}}{\partial \xi}\frac{\partial \xi}{\partial x_{1,l}} + \frac{\partial P_{1,j}}{\partial \eta}\frac{\partial \eta}{\partial x_{1,l}}, \tag{32}$$

$$\frac{\partial P_{1,j}}{\partial y_{1,l}} = \frac{\partial P_{1,j}}{\partial \xi}\frac{\partial \xi}{\partial y_{1,l}} + \frac{\partial P_{1,j}}{\partial \eta}\frac{\partial \eta}{\partial y_{1,l}}. \tag{33}$$

As pointed out before, the x-coordinate of the grid points are considered fixed and they are not included in the design variables. Thus (32) is written here to derive the required relations for the sensitivity coefficients. By interchanging x and ξ, and y and η, and solving the derived equations for $\partial P/\partial x$ and $\partial P/\partial y$, we finally obtain

$$\frac{\partial P_{1,j}}{\partial y_{1,l}} = \frac{1}{J}\left[-\left(x_\eta\right)_{1,l}\left(P_\xi\right)_{1,j} + \left(x_\xi\right)_{1,l}\left(P_\eta\right)_{1,j}\right], \tag{34}$$

where $J = (x_\xi y_\eta - x_\eta y_\xi)_{1,l}$ is the Jacobian of the transformation. Using the finite difference method to discretize the equations in the computational domain, we can write appropriate algebraic approximations for all partial derivatives involved in the above equation. Therefore,

$$\left(P_\xi\right)_{1,j} = \frac{-3P_{1,j} + 4P_{2,j} - P_{3,j}}{2}, \tag{35}$$

$$\left(P_\eta\right)_{1,j} = \frac{P_{1,j+1} - P_{1,j-1}}{2}, \tag{36}$$

$$\left(x_\xi\right)_{1,l} = \frac{-3x_{1,l} + 4x_{2,l} - x_{3,l}}{2}, \tag{37}$$

$$\left(x_\eta\right)_{1,l} = \frac{x_{1,l+1} - x_{1,l-1}}{2} \tag{38}$$

which are based on the central and the forward differences. Equations (35) through (38) are employed to calculate the sensitivity coefficients in (31).

Bezier Control Points as Design Variables. With regard to (27) and considering the control points of the Bezier curve as design variable, we can write

$$\frac{\partial \mathcal{J}}{\partial B_{y_l}} = 2 \sum_{i=1}^{2m-4} \left(P_{iB} - P_{iB_d}\right) \frac{\partial P_{iB}}{\partial B_{y_l}}. \tag{39}$$

Using the chain rule, we can write

$$\frac{\partial P_{iB}}{\partial B_{y_l}} = \frac{\partial P_{iB}}{\partial y_{i'B}}\frac{\partial y_{i'B}}{\partial B_{y_l}}, \tag{40}$$

where $y_{i'B}$ ($i' = 1,\ldots,2m-4$) are the y-coordinate of the predetermined grid points to be passed by the Bezier curve and B_{y_l} ($l = 1,\ldots,18$) are the y-coordinate of Bezier control points whose number is equal to 18 (9 for each of the upper and lower surfaces). The term $\partial P_{iB}/\partial y_{i'B}$ can be computed by the expressions derived for Case 1 (see (31)). The size of the matrix formed by the arrays $\partial P_{iB}/\partial y_{i'B}$ is $(2m-4) \times (2m-4)$. The term $\partial y_{i'B}/\partial B_{y_l}$ can be easily evaluated by taking derivative of (19) with respect to the control points B (noting that $[P(t)] \equiv [x(t), y(t)]$). The control points may be renumbered so that $B_{y_1} = B_{y_{9U}}$, $B_{y_2} = B_{y_{8U}},\ldots,$ $B_{y_9} = B_{y_{1U}}$ and $B_{y_{10}} = B_{y_{1L}}$, $B_{y_{11}} = B_{y_{2L}},\ldots,$ $B_{y_{18}} = B_{y_{9L}}$. The indices U and L denote the upper and lower surfaces, respectively. The direction of numbering is from right to left for the upper surface and from left to right for the lower surface. The reason for this renumbering is the compatibility with the grid point data reading (most of the airfoil data are in this format) as well as the pressure reading to compute the objective function (see (27)). However, we should note that the Bezier curve evaluation is from left to right for both the upper and lower surfaces. The size of the matrix formed by the arrays $\partial y_{i'B}/\partial B_{y_l}$ is $(2m-4) \times 18$. Because the upper and lower surfaces are constructed separately, the variation of y of the upper surface with respect to the change in position of the lower surface control points as well as the variation of y

of the lower surface with respect to the change in position of the upper surface control points is zero.

Maximum Thickness as Design Variables. In a similar derivation to Case 1, the sensitivity matrix for Case 3 will be

$$\mathbf{Ja} = \begin{bmatrix} \dfrac{\partial P_1}{\partial y_1} & \dfrac{\partial P_1}{\partial y_2} \\ \dfrac{\partial P_2}{\partial y_1} & \dfrac{\partial P_2}{\partial y_2} \end{bmatrix}. \tag{41}$$

7. Optimization Method

In this paper, two powerful optimization methods, namely, the conjugate gradient method and the quasi-Newton method will be used. For the airfoil grid points as a design variable (Case 1) both optimization methods will be employed. However, for the Bezier curve control points as a design variable (Case 2), only the quasi-Newton method will be used. For Case 3 (the maximum thickness of NACA00xx airfoils as a design variable), only the conjugate gradient method will be employed.

Conjugate Gradient Method. The conjugate gradient algorithm to obtain the optimal shape for the airfoil is as follows.

(1) Specify the physical domain, the boundary conditions, the problem conditions such as Mach number and the angle of attack, and the desired airfoil surface pressure distribution.

(2) Generate the boundary fitted grids using the grid generation methods described earlier.

(3) Solve the direct flow problem of finding the pressure values at any grid points of the physical domain and hence the airfoil surface.

(4) Using (26), compute the objective function ($\mathcal{F}^{(k)}$).

(5) If the value of the objective function obtained in step (4) is less than the specified stopping criterion, the optimization is finished. Otherwise, go to step (6).

(6) Compute the sensitivity matrix (**Ja**) from (31).

(7) Compute the gradient direction $\nabla\mathcal{F}^{(k)}$ from (30).

(8) Compute the conjugation coefficient $\gamma^{(k)}$ from the following equation (the Polak-Ribiere formula):

$$\begin{aligned} \gamma^{(k)} &= \frac{\left[\nabla\mathcal{F}^{(k)}\right]^T\left(\nabla\mathcal{F}^{(k)} - \nabla\mathcal{F}^{(k-1)}\right)}{\left\|\nabla\mathcal{F}^{(k-1)}\right\|^2} \\ &= \frac{\left[\nabla\mathcal{F}^{(k)}\right]^T\left(\nabla\mathcal{F}^{(k)} - \nabla\mathcal{F}^{(k-1)}\right)}{\left[\nabla\mathcal{F}^{(k-1)}\right]^T\nabla\mathcal{F}^{(k-1)}}. \end{aligned} \tag{42}$$

For $k = 0$, set $\gamma^{(0)} = 0$.

(9) Compute the direction of descent $\mathbf{d}^{(k)}$ from the following:

$$\mathbf{d}^{(k)} = \nabla\mathcal{F}^{(k)} + \gamma^{(k)}\mathbf{d}^{(k-1)}. \tag{43}$$

(10) Compute the search step size $\beta^{(k)}$ from the following:

$$\beta^{(k)} = \frac{\left[\mathbf{Ja}^{(k)}\mathbf{d}^{(k)}\right]^T\left[P_{1,j} - P_{d(1,j)}\right]}{\left[\mathbf{Ja}^{(k)}\mathbf{d}^{(k)}\right]^T\left[\mathbf{Ja}^{(k)}\mathbf{d}^{(k)}\right]}. \tag{44}$$

(11) Evaluate the new y-coordinates of the airfoil surface grid nodes as follows:

$$\mathbf{y}^{(k+1)} = \mathbf{y}^{(k)} - \beta^{(k)}\mathbf{d}^{(k)}. \tag{45}$$

(12) Set the next iteration ($k = k + 1$) and return to step (2).

The above algorithm is for the airfoil grid points as a design variable (Case 1) only. The algorithm for Case 3 can be expressed in a similar way.

Quasi-Newton Method. Quasi-Newton method is another powerful optimization method used in this paper. In quasi-Newton method, the Hessian matrix (which is composed of the second partial derivatives) is replaced by an approximation of it. The approximation uses only the first partial derivatives. The *Broyden-Fletcher-Goldfarb-Shanno* (BFGS) method is a quasi-Newton method for solving unconstrained nonlinear optimization. In the BFGS method, the Hessian matrix approximation, $\mathbf{B}^{(k)}$, is updated iteratively. The steps of BFGS method can be summarized as follows.

(1) Specify the physical domain, the boundary conditions, the problem conditions such as Mach number and the angle of attack, and the desired airfoil surface pressure distribution.

(2) Generate the boundary fitted grids using the grid generation methods described earlier.

(3) Solve the direct flow problem of finding the pressure values at any grid points of the physical domain and hence the airfoil surface.

(4) Using (26), compute the objective function ($\mathcal{F}^{(k)}$).

(5) If value of the objective function obtained in step (4) is less than the specified stopping criterion, the optimization is finished. Otherwise, go to step (6).

(6) Compute the sensitivity matrix (**Ja**) from (31).

(7) Compute the gradient direction $\nabla\mathcal{F}^{(k)}$ from (30).

(8) The initial Hessian matrix approximation, $\mathbf{B}^{(1)}$, is taken as the identity matrix, namely, $\mathbf{B}^{(1)} = \mathbf{I}$.

(9) Set $\mathbf{S}^{(k)} = -\mathbf{B}^{(k)}\nabla\mathcal{F}^{(k)}$ and the iteration number as $k = 1$.

(10) Compute the search step size $\beta^{(k)}$ (from (44)) in the direction $\mathbf{S}^{(k)}$ and set

$$\mathbf{y}^{(k+1)} = \mathbf{y}^{(k)} - \beta^{(k)}\mathbf{S}^{(k)}. \tag{46}$$

(11) Repeat the steps (2) to (7) with these new values of $\mathbf{y}$ for the grid points y-coordinates to calculate $\nabla\mathcal{F}^{(k+1)}$.

Table 1: Data used for Test Case 1.

	Airfoil	Grid size	Angle of attack α	Free stream velocity V_∞
Initial	TsAGI"B" 10%	300 × 305	1°	70 m/s
Desired	NACA 0015	350 × 365	1°	70 m/s

(12) Update the Hessian matrix approximation as

$$\mathbf{B}^{(k+1)} = \mathbf{B}^{(k)} + \left(1 + \frac{\left(\mathbf{g}^{(k)}\right)^T \mathbf{B}^{(k)} \mathbf{g}^{(k)}}{\left(\mathbf{d}^{(k)}\right)^T \mathbf{g}^{(k)}}\right) \frac{\mathbf{d}^{(k)} \left(\mathbf{d}^{(k)}\right)^T}{\left(\mathbf{d}^{(k)}\right)^T \mathbf{g}^{(k)}} - \frac{\mathbf{d}^{(k)} \left(\mathbf{g}^{(k)}\right)^T \mathbf{B}^{(k)}}{\left(\mathbf{d}^{(k)}\right)^T \mathbf{g}^{(k)}} - \frac{\mathbf{B}^{(k)} \mathbf{g}^{(k)} \left(\mathbf{d}^{(k)}\right)^T}{\left(\mathbf{d}^{(k)}\right)^T \mathbf{g}^{(k)}}, \tag{47}$$

where

$$\mathbf{d}^{(k)} = \mathbf{y}^{(k+1)} - \mathbf{y}^{(k)} = -\beta^{(k)} \mathbf{S}^{(k)},$$
$$\mathbf{g}^{(k)} = \nabla \mathcal{J}^{(k+1)} - \nabla \mathcal{J}^{(k)}. \tag{48}$$

(13) Set the new iteration number as $k = k + 1$ and go to step (9).

8. Results

In this section, the results obtained for the shape optimization of an airfoil in the incompressible, irrotational, and inviscid flow under given boundary conditions are presented. Three kinds of the design variable (the airfoil grid points, the Bezier curve control points, and the maximum thickness of NACA 00xx airfoils) as well as two optimization methods (CG and BFGS) are considered. In all test cases in this paper which employ the Bezier curve, the number of predetermined airfoil data, m, is set equal to the Bezier curve order, $n + 1$.

Test Case 1. In this test case, the airfoil surface is parameterized by a Bezier curve of order 11. The total number of the design variable is 18, namely, 9 design variables for each of the upper and lower surfaces. At first, two parametric curves for two surfaces (upper and lower) are obtained using 11 grid points and then a fine grid is generated to obtain accurate results. The data for Test Case 1 is given in Table 1. The comparison of the initial and optimal airfoil shapes and some magnified parts of them are shown in Figures 15, 20, and 26. In this test case, BFGS optimization method is employed.

The convergence of the objective function is shown in Figures 17, 23, 29, 37, and 39. The initial and minimum values for the objective function are approximately 3517433 and 2877116, respectively, which shows %18.2 reduction in objective function (see Figures 16, 21, and 27). The minimum value for the objective function takes place in iteration 14. The optimization time spent on the 1st iteration (which is equivalent to one direct flow solution) is 11 minutes and 43 seconds and the total optimization time for 30 iterations is 11 minutes and 46 seconds which shows the proposed sensitivity analysis efficiency. 30 iterations take only 3 seconds. The reason for the difference between the 1st iteration and the following ones is that the solution after the 1st iteration is a very good initial guess for the 2nd iteration and the direct solution converges quickly. In other words, what is a bit time consuming for the 1st iteration is the grid generation and stream function loops not the pressure calculation, sensitivity analysis, and optimization stages. Moreover, a fine

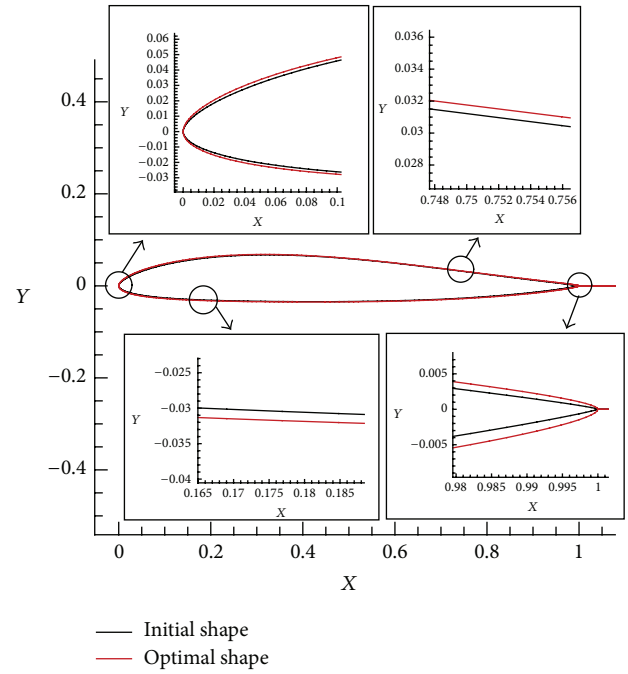


Figure 15: Comparison of the initial and optimal shapes and some magnified parts including the leading edge, middle parts, and the trailing edge.

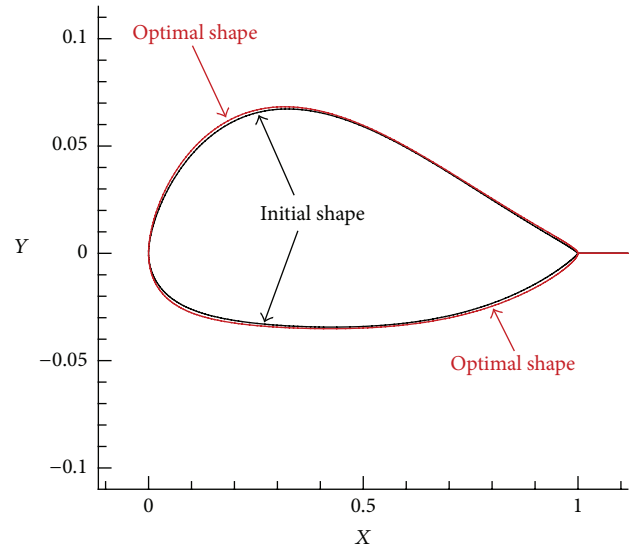


Figure 16: Comparison of the initial and optimal shapes. The y-axis has been greatly exaggerated to highlight difference in the airfoil shapes.

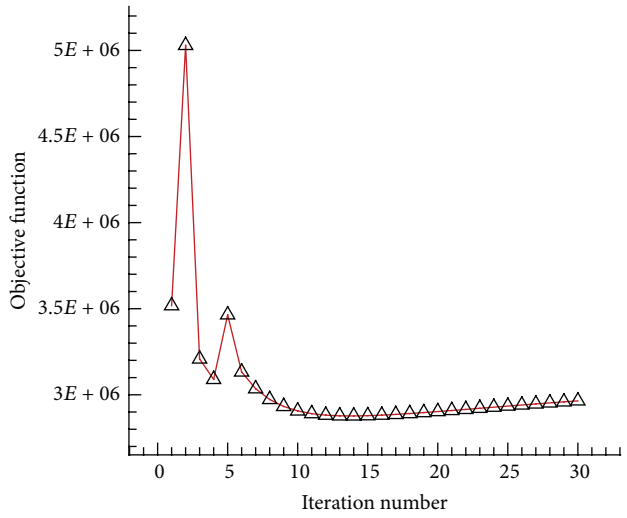


FIGURE 17: Objective function value versus the iteration number.

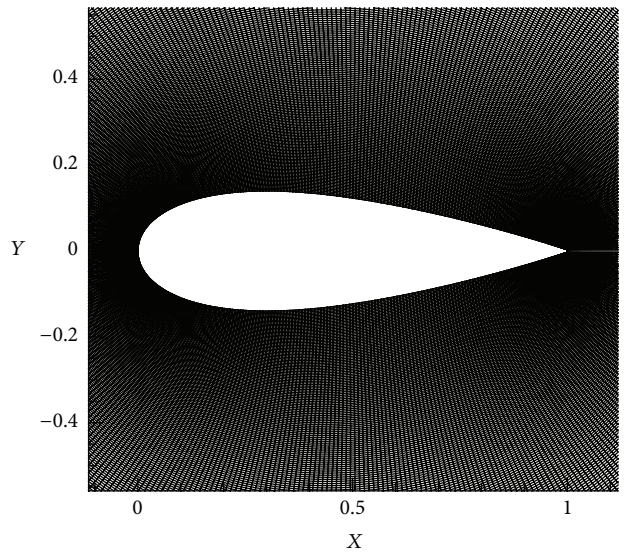


FIGURE 18: Grid used in Test Case 2 (around initial shape).

grid (300×305) and a tolerance of 10^{-8} are used in the iterative loops which increase the computation time. The code is programmed in Fortran 77 using a Fortran compiler (Force 2.0) and the computations are run on a PC with Intel Pentium Dual 1.73 and 1 G RAM. All the computations in the test cases in this paper are performed using the above mentioned compiler and PC. Therefore, there is no need to repeat it in the following test cases.

Test Case 2. Test Case 2 is similar to Test Case 1 but with different specifications (see Figure 18). The data for this test case is given in Table 2.

The explanation is similar to Test Case 1. Thus only the results will be given in Table 3.

Now an optimal shape design problem using a Bezier curve of order 7 is given. As it will be shown, it decreases the objective function value much bigger than when using

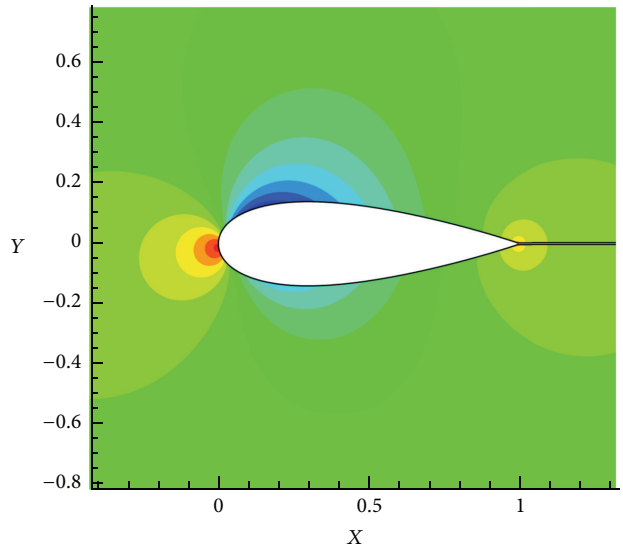


FIGURE 19: Pressure distribution around the airfoil surface (initial shape).

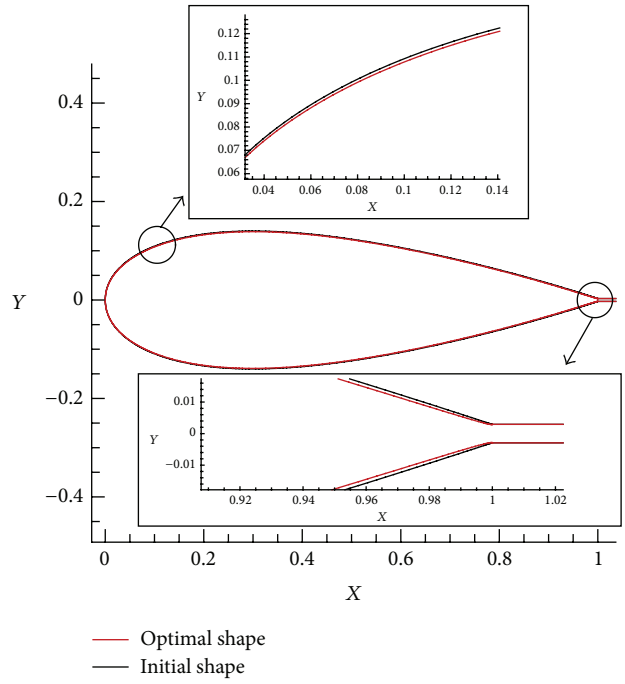


FIGURE 20: Comparison of the initial and optimal shapes and some magnified parts including the leading and trailing edges.

TABLE 2: Data used for Test Case 2.

	Airfoil	Grid size	Angle of attack α	Free stream velocity V_∞
Initial	NACA 0028	400×425	2°	70 m/s
Desired	NACA 0016	350×365	2°	70 m/s

a Bezier curve of order 11 as there is less number of design variables (10 design variables for a Bezier curve of order 7

Table 3: Results for Test Case 2.

Initial objective function	Minimum objective function (iteration 3)	Computational time, 1st iteration	Total computational time, 30 iterations	Percentage of reduction in objective function
10074507	5585758	35 m and 30 s	35 m and 34 s	44.5%

Table 4: Data used for Test Case 3.

	Airfoil	Grid size	Angle of attack α	Free stream velocity V_∞
Initial	NACA 0012	300 × 305	0°	70 m/s
Desired	NACA 0017	250 × 305	0°	70 m/s

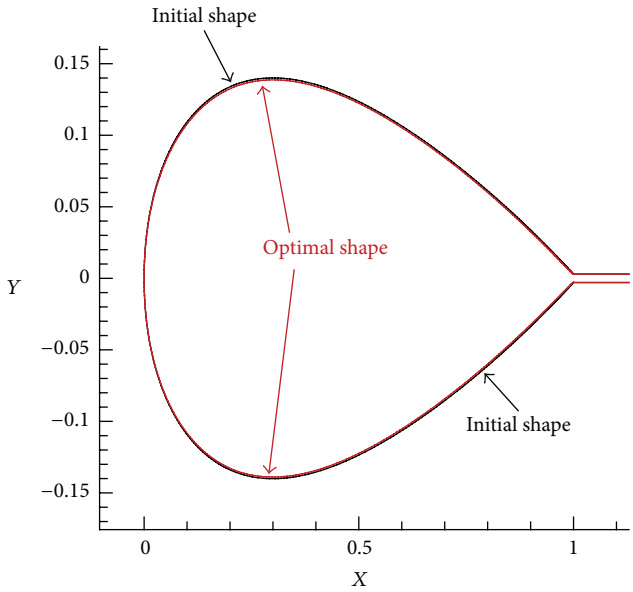


Figure 21: Comparison of the initial and optimal shapes. The y-axis has been greatly exaggerated to highlight difference in the airfoil shapes.

versus 18 design variables for a Bezier curve of order 11). However, as it will be shown, using a Bezier curve of order 7 is not comprehensive for all airfoil shapes and is suitable to NACA 00XX or similar airfoils only. In other words, it is not able to produce all airfoil shapes precisely.

Test Case 3 (using a Bezier curve of order 7; see Figure 24 and Table 4). The results are given in Table 5.

Although there is a decrease of %59 in objective function, the %59 approach from the initial shape to desired one is not seen (see Figure 28). This indicates that the solution of the inverse problem is not unique. The reason for this can be found in the trailing edge configuration for the initial, optimal, and desired airfoil shapes (Figure 28).

Although the results of using the Bezier curve of order 7 is very promising, its drawback is that it is restricted to the simple and symmetric airfoil shapes such as NACA 00xx. For other airfoil shapes, there can be seen some oscillations around the trailing edge (see Figure 30).

Moreover, the Bezier curve of order 7 fails to represent the NACA 00xx airfoils accurately. In other words, the Bezier curve of order 11 is the appropriate option to produce very accurate airfoils (Figure 31).

Test Case 4. In this test case, the airfoil surface is parameterized by grid points obtained from the analytical NACA formula (e.g., software JavaFoil). In this case, the number of the design variables is equal to the number of grid points minus three (one for leading edge and two for trailing edge). Therefore, we have an aerodynamic shape optimization problem with a high number of the design variables as we should have a fine grid to obtain sufficiently accurate results. It is known that the optimization process becomes more challenging by increasing the number of design variables. Hence we have a difficult shape optimization problem in Test Case 4. The data used for Test Case 4 is given in Table 6. A very fine grid (400 × 425) is used for the initial airfoil shape (NACA 0012). The number of the design variables is $N - 3$ which is $425 - 3 = 422$. Therefore, a time consuming optimization problem is expected. However, by using the sensitivity analysis scheme proposed in this paper, the total time for the optimization problem in Test Case 4 using both CG and BFGS optimization methods is about 46 minutes for 8000 iterations. The computation time for the 1st iteration is about 25 minutes. The comparison of the computation times for the 1st iteration and 8000 ones indicates again the efficiency of the sensitivity analysis scheme. The summary of the results is presented in Table 7. The comparison of the initial, optimal, and desired airfoil shapes is given in Figure 32. As can be seen in the figure, the variation of the shape is minute. The convergence of the objective function to a local minimum when both the CG and BFGS optimization methods are used as well as a comparison of them is shown in Figures 33, 34, and 35, respectively. The plots reveal the better performance of the BFGS method in minimizing the objective function.

In Test Cases 5 and 6 the maximum thickness of the NACA 00xx is considered as a design variable. As mentioned previously, the conjugate gradient method is employed as the optimization method.

Test Case 5. The data for the problem including the conditions for the initial and desired airfoils are given in Table 8.

Figure 36 represents the comparison of the initial, optimal, and desired shapes for airfoils (also see Figures 22 and 38). The desired airfoil shape is a NACA0018 at conditions stated in Table 8. As can be seen, this shape is shown by

Table 5: Results for Test Case 3.

Initial objective function	Minimum objective function (iteration 34)	Computational time, 1st iteration	Total computational time, 100 iterations	Percentage of reduction in objective function
898639	368148	9 m and 32 s	9 m and 43 s	59%

Table 6: Data used for Test Case 4.

	Airfoil	Grid size	Angle of attack α	Free stream velocity V_∞
Initial	NACA 0012	400 × 425	0°	70 m/s
Desired	NACA 0014	400 × 425	0°	70 m/s

Table 7: Summary of the results of Test Case 4.

	Initial $\mathcal{J}$	Final $\mathcal{J}$	Number of iterations	Computation time (total)	Reduction in $\mathcal{J}$
CG	6351997	6053996	8000	~46 mins	4.7%
BFGS	6351997	6036943	8000	~46 mins	5%

Table 8: Data used for Test Case 5.

	Airfoil	Grid size	Angle of attack α	Free stream velocity V_∞
Initial	NACA 0011	80 × 125	2°	70 m/s
Desired	NACA 0018	80 × 125	2°	70 m/s

a black color line. The optimization process is started by a NACA0011 as an initial shape which is shown by a red color line. The optimal shape (shown by a blue color line), which is obtained by the conjugate gradient method, is in an excellent agreement (full matching) with the desired one. The objective function variation is shown in Figure 37. The initial and final values for the objective function are about 773597.45 and 15.48, respectively, which reveals an approximately %100 reduction in the objective function within 31 iterations. The total time for the optimization (for 31 iterations) is 2 minutes and 14 seconds. The tolerance used in iterative steps in the program is 10^{-8}. Although such a tolerance value increases the computation time, it enhances the accuracy of the results. If the y-components of the maximum thickness in upper and lower airfoil surfaces are denoted by $y_{\text{thick}U}$ and $y_{\text{thick}L}$, respectively, the value of the pressure for these two locations for initial, optimal, and desired shapes are reported in Tables 10 and 13. The difference values show the validity of the shape optimization process.

Test Case 6. The data for the problem is given in Table 11. The explanation for the results is similar to Test Case 5 (see Tables 9 and 12).

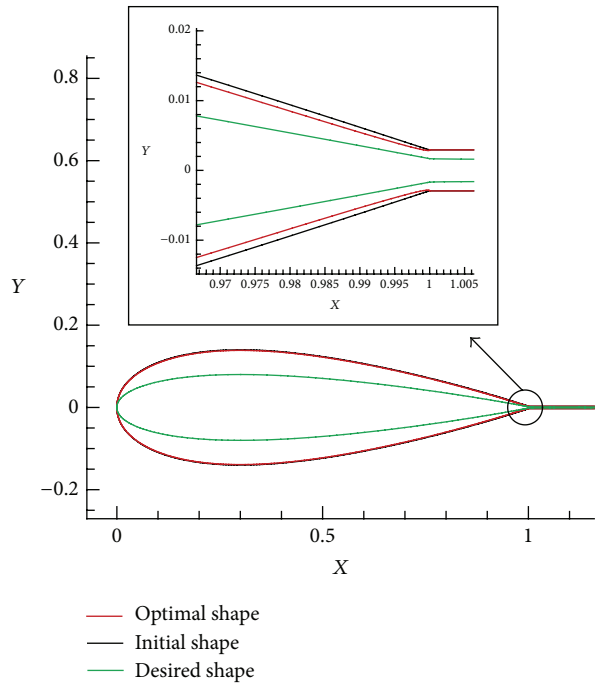


Figure 22: Comparison of the initial, optimal, and desired shapes.

Table 9: Results for Test Case 5.

	Initial $\mathcal{J}$	Final $\mathcal{J}$	Number of iterations	Computation time (total)	Reduction in $\mathcal{J}$
CG	773597.45	15.48	31	2 m : 14 s	~100%

9. Adjoint Method

As pointed out previously, for the aerodynamic shape optimization problems requiring a large number of design variables, the use of finite difference method to evaluate the gradient by introducing a small perturbation to each design variable separately and then solving the flow problem is of very high computational cost, because it requires a number of additional flow solutions equal to the number of design variables. For optimal shape design problems with a high number of design variables, the adjoint method [4] can compute the gradients of objective function much faster than the finite difference method.

The aerodynamic shape optimization problem of interest here can be expressed as

$$\begin{aligned} &\text{minimization of objective function} && \mathcal{J} \\ &\text{subject to constraint} && \mathcal{R} = 0 \ \text{(the governing equation)}. \end{aligned} \tag{49}$$

Table 10: Comparison of the pressure at the maximum thicknesses of the airfoil surface (upper and lower surfaces) for the initial, optimal, and desired shapes.

	Initial shape	Optimal shape	Desired shape	Difference (Pa)
Pressure at y_{thickU} (Pa)	100080.04	99411.37	99410.06	Difference in initial and desired = 669.98 Difference in optimal and desired = 1.31
Pressure at y_{thickL} (Pa)	100814.39	100248.25	100244.54	Difference in initial and desired = 569.85 Difference in optimal and desired = 3.71

Table 11: Data used for Test Case 6.

	Airfoil	Grid size	Angle of attack α	Free stream velocity V_∞
Initial	NACA 0015	80×125	2°	70 m/s
Desired	NACA 0035	80×125	2°	70 m/s

Table 12: Results for Test Case 6.

	Initial $\mathcal{J}$	Final $\mathcal{J}$	Number of iterations	Computation time (total)	Reduction in $\mathcal{J}$
CG	7986880.34	1.15	21	1 m : 08 s	~100%

Table 13: Comparison of the pressure at the maximum thicknesses of the airfoil surface (upper and lower surfaces) for the initial, optimal, and desired shapes.

	Initial shape	Optimal shape	Desired shape	Difference (Pa)
Pressure at y_{thickU} (Pa)	99703.83	97552.39	97552.99	Difference in initial and desired = 2150.84 Difference in optimal and desired = 0.60
Pressure at y_{thickL} (Pa)	100494.52	98662.17	98661.28	Difference in initial and desired = 1833.24 Difference in optimal and desired = 0.89

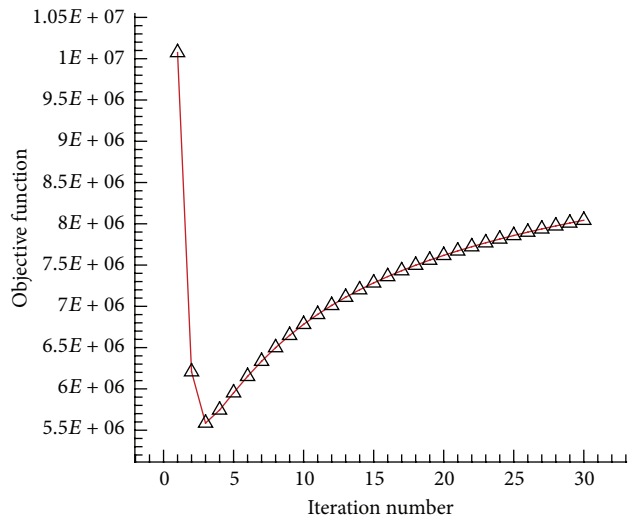


Figure 23: Objective function value versus the iteration number.

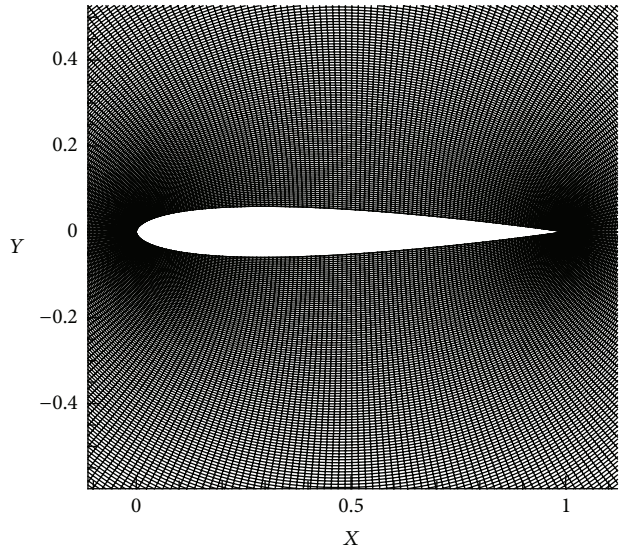


Figure 24: Grid used in Test Case 3 (around initial shape).

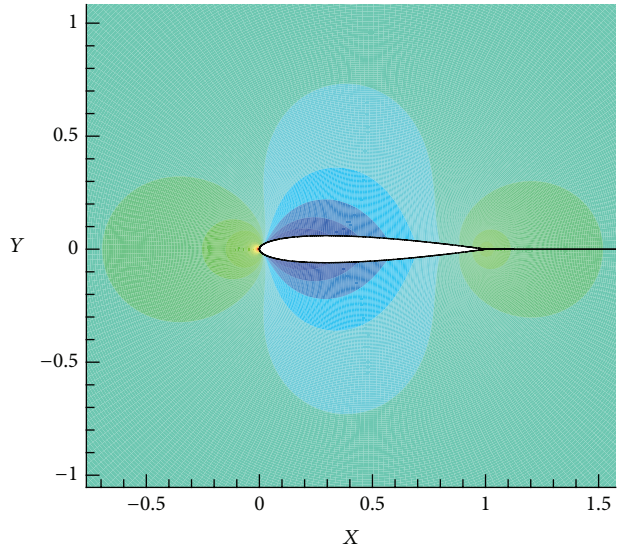


Figure 25: Pressure distribution around the airfoil surface (initial shape).

The objective function $\mathcal{J}$ and the governing equation $\mathcal{R} = 0$ depend on the flow variables $\mathbf{W}$ and the geometry design variable $\mathbf{X}_D$:

$$\begin{aligned} \mathcal{J} &= \mathcal{J}(\mathbf{W}, \mathbf{X}_D), \\ \mathcal{R} &= \mathcal{R}(\mathbf{W}, \mathbf{X}_D) = 0. \end{aligned} \tag{50}$$

The derivative of the objective function $\mathcal{J}$ with respect to the design variables $\mathbf{X}_D$ can be expressed as

$$\frac{d\mathcal{J}}{d\mathbf{X}_D} = \frac{\partial \mathcal{J}}{\partial \mathbf{X}_D} + \frac{\partial \mathcal{J}}{\partial \mathbf{W}} \frac{\partial \mathbf{W}}{\partial \mathbf{X}_D} \tag{51}$$

which states that a change in the objective function is due to a combination of a variation in the flow solution $\partial\mathbf{W}$ and a variation in the design variable (change in geometry) $\partial\mathbf{X}_D$. In a similar way, we have

$$\frac{d\mathcal{R}}{d\mathbf{X}_D} = \frac{\partial \mathcal{R}}{\partial \mathbf{X}_D} + \frac{\partial \mathcal{R}}{\partial \mathbf{W}} \frac{\partial \mathbf{W}}{\partial \mathbf{X}_D} = 0. \tag{52}$$

If the sensitivity analysis is performed using (51) and (52), the problem is referred to as the "primal problem." Solving the primal problem comes with the same difficulties as we encounter with use of the finite difference method. It requires the additional flow solutions proportional to the number of the design variables $\mathbf{X}_D$. Therefore, the adjoint method comes to the picture by introducing a vector of Lagrange multipliers

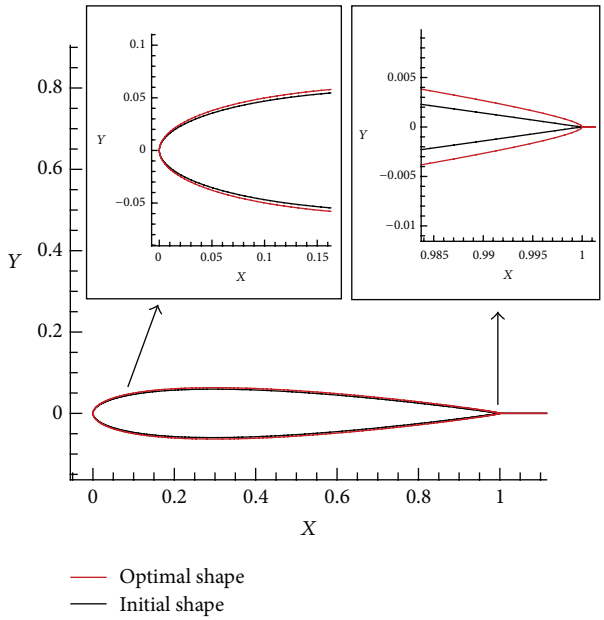


Figure 26: Comparison of the initial and optimal shapes and some magnified parts including the leading and trailing edges.

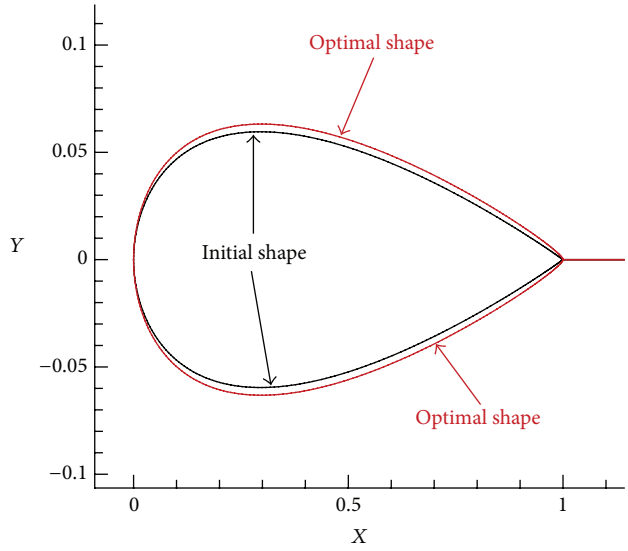


Figure 27: Comparison of the initial and optimal shapes. The y-axis has been greatly exaggerated to highlight difference in the airfoil shapes.

Ψ. By adding (52) as a constraint to the sensitivity equation (51), we obtain

$$\frac{d\mathcal{J}}{d\mathbf{X}_D} = \frac{\partial \mathcal{J}}{\partial \mathbf{X}_D} + \frac{\partial \mathcal{J}}{\partial \mathbf{W}} \frac{\partial \mathbf{W}}{\partial \mathbf{X}_D} - \Psi^T \left\{ \overbrace{\frac{\partial \mathcal{R}}{\partial \mathbf{X}_D} + \frac{\partial \mathcal{R}}{\partial \mathbf{W}} \frac{\partial \mathbf{W}}{\partial \mathbf{X}_D}}^{=0} \right\}. \tag{53}$$

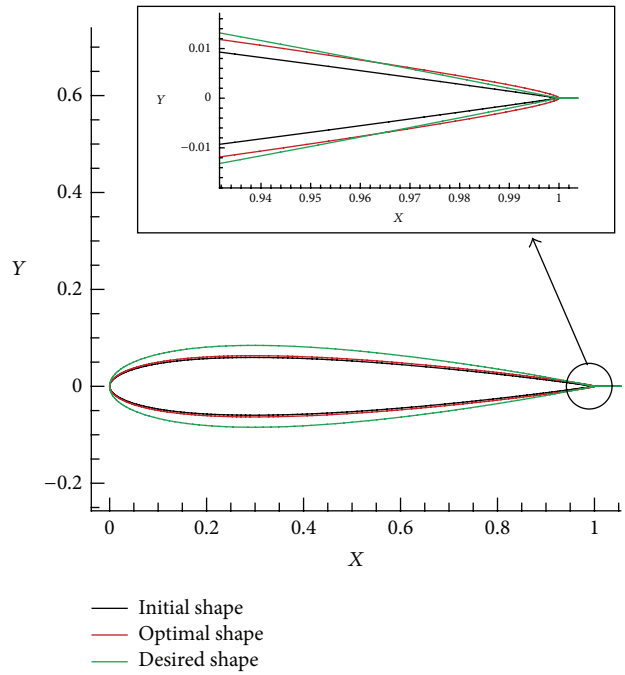


Figure 28: Comparison of the initial, optimal, and desired shapes.

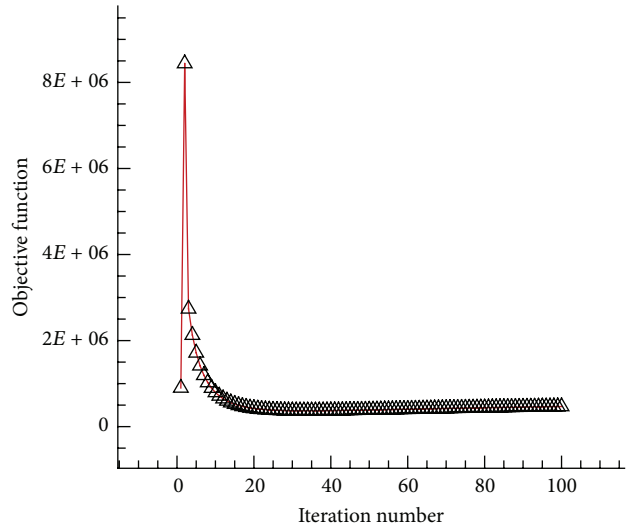


Figure 29: Objective function value versus the iteration number.

Rearranging the terms inside (53), we get

$$\frac{d\mathcal{J}}{d\mathbf{X}_D} = \left(\frac{\partial \mathcal{J}}{\partial \mathbf{X}_D} - \Psi^T \frac{\partial \mathcal{R}}{\partial \mathbf{X}_D} \right) + \left(\frac{\partial \mathcal{J}}{\partial \mathbf{W}} - \Psi^T \frac{\partial \mathcal{R}}{\partial \mathbf{W}} \right) \frac{\partial \mathbf{W}}{\partial \mathbf{X}_D}. \tag{54}$$

If

$$\frac{\partial \mathcal{J}}{\partial \mathbf{W}} - \Psi^T \frac{\partial \mathcal{R}}{\partial \mathbf{W}} = 0, \tag{55}$$

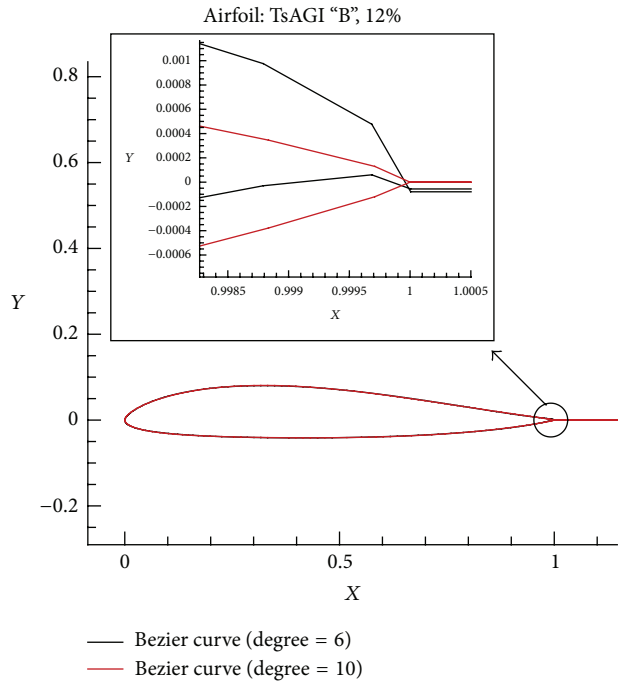


Figure 30: Oscillations around the trailing edge.

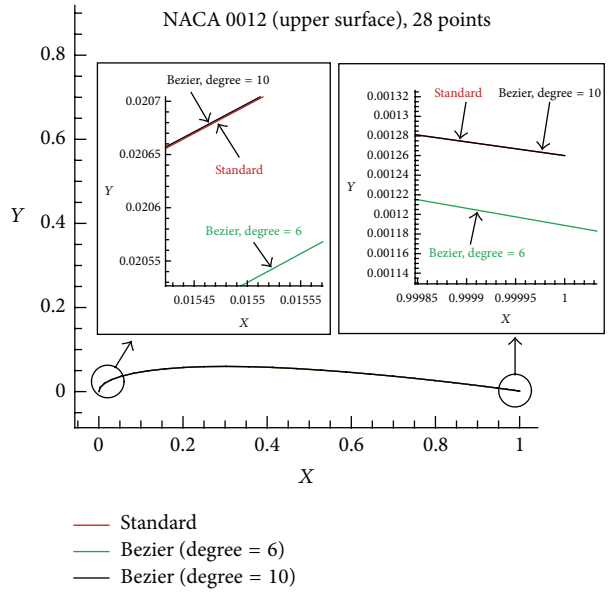


Figure 31: Comparison of an analytical NACA 0012 airfoil (upper surface only) with one obtained by using the Bezier curves of orders 7 and 11. The plots represent an excellent agreement between the analytical NACA and the Bezier of order 11.

then (54) reduces to

$$\frac{d\mathcal{J}}{d\mathbf{X}_D} = \frac{\partial \mathcal{J}}{\partial \mathbf{X}_D} - \Psi^T \frac{\partial \mathcal{R}}{\partial \mathbf{X}_D}. \tag{56}$$

Equation (55) is the *adjoint equation* and the vector Ψ is the *adjoint variables*. Equations (55) and (56) are referred to as the "dual problem." The adjoint equation is a linear system

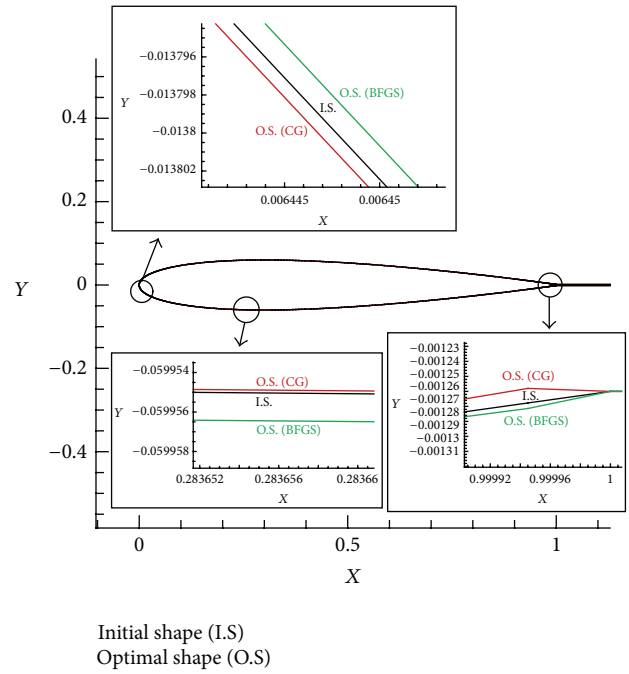


Figure 32: Comparison of the initial and optimal airfoil shapes with the magnified sections of them to show the variation of the shape. Both BFGS and CG are used in optimization process.

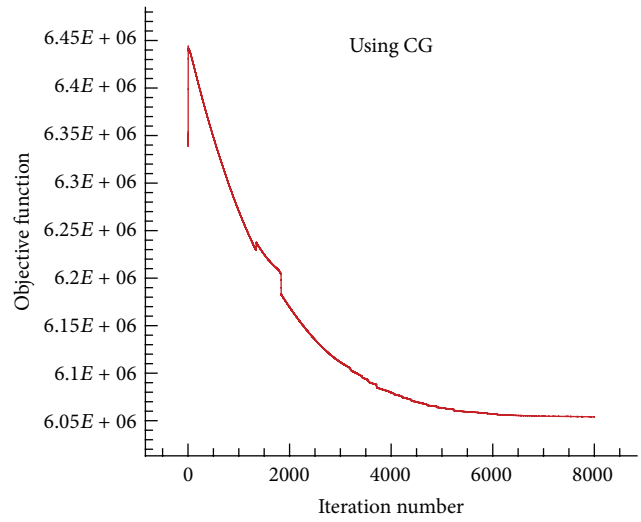


Figure 33: Convergence of the objective function. CG is used as the optimization method.

and can be solved to obtain Ψ. Then the determined Ψ can be substituted into (56) to obtain the gradient of the objective function. It can be seen that the gradient of the objective function can be determined without the need for additional flow solutions. The computational cost of solving the adjoint equation is comparable to that of solving the flow equation. Therefore, the computational cost of evaluating the objective function gradient is roughly equal to the computational cost of two flow equation solutions, independent of the number of design variables [38–41].

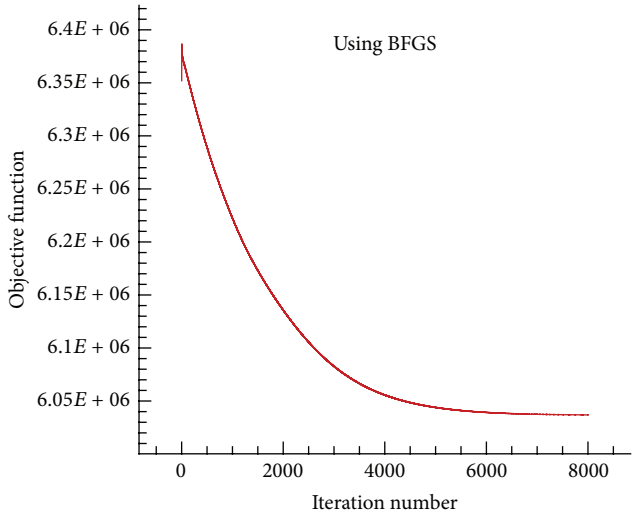


Figure 34: Convergence of the objective function. BFGS is used as the optimization method.

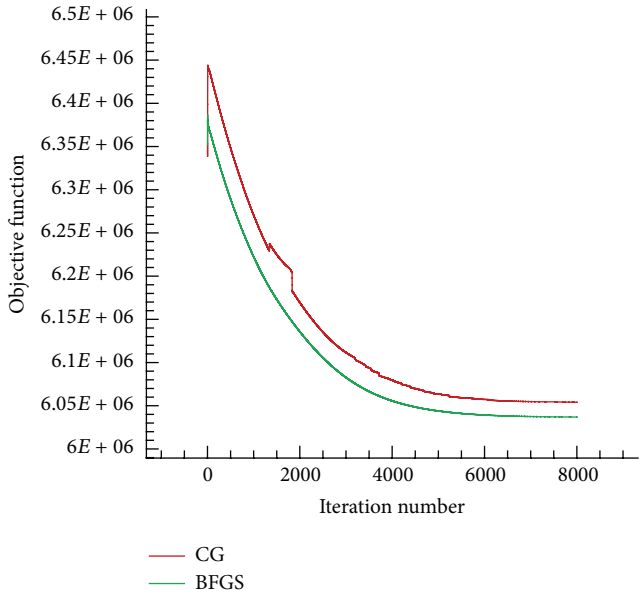


Figure 35: Comparison of the CG and BFGS methods in decreasing the objective function.

From the accuracy of the derivatives view point, the finite difference method (based on the perturbation scheme) is compared to the adjoint method [42–44]. The comparison shows a very good agreement between two methods. Therefore, our aim here is to compare our novel shape sensitivity method to the adjoint method from the efficiency view point only. As mentioned above, the computational cost of solving the adjoint equation is comparable to that of solving the flow equation whereas the computational cost of our novel method is comparable to that of computation of an algebraic expression for arrays of a matrix. As seen in Test Case 4, the computation time for iterations 2 to 8000 is 46 − 25 = 21 minutes (about 7 iterations per second) which reveals the efficiency of the proposed sensitivity analysis.

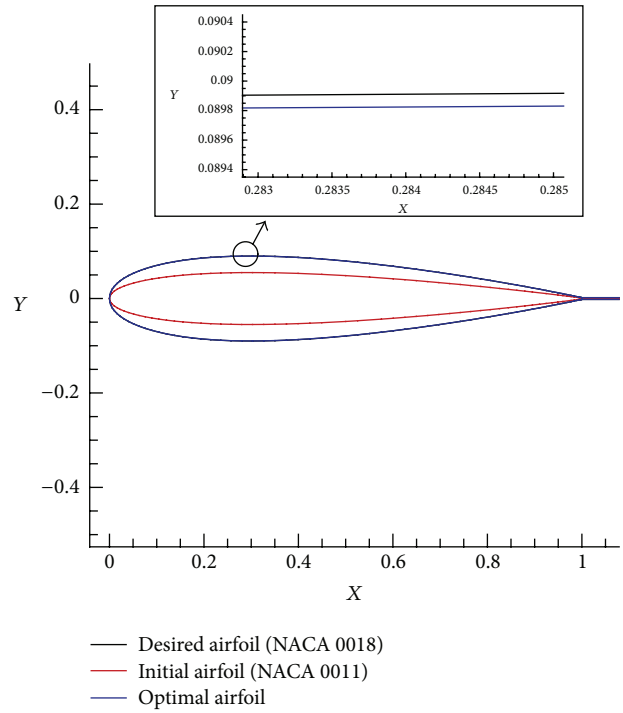


Figure 36: The initial, optimal, and desired shapes for the airfoil. There is an excellent agreement between the optimal and desired airfoil shapes.

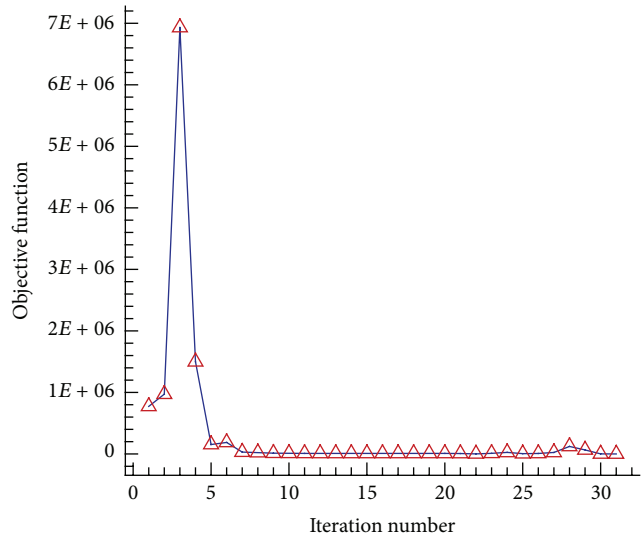


Figure 37: Objective function value versus the iteration number.

10. Conclusion

This paper addressed the aerodynamic shape optimization for an airfoil in an irrotational and incompressible flow governed

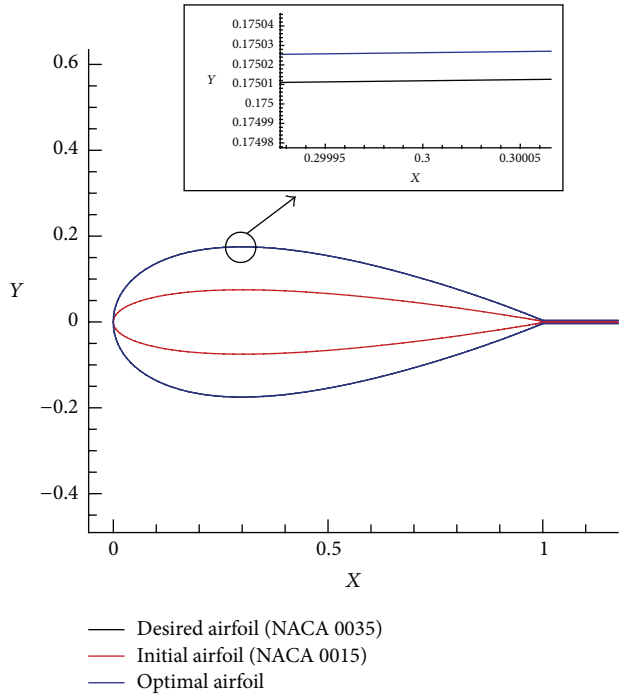


Figure 38: The initial, optimal, and desired shapes for the airfoil. There is an excellent agreement between the optimal and desired airfoil shapes.

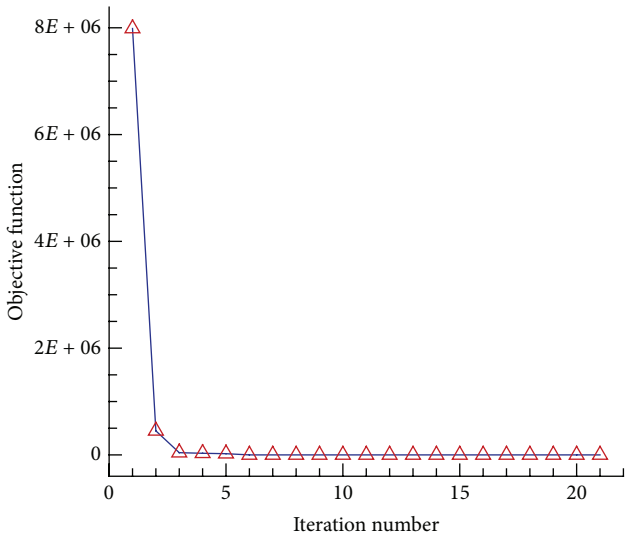


Figure 39: Objective function value versus the iteration number.

by Laplace equation using a type of the elliptic grid generation (O-type), a novel and very efficient sensitivity analysis method, and the conjugate gradient and BFGS optimization methods. The airfoil was parameterized using the grid points and the Bezier curve. Three different types of design variable were considered: the grid points, the Bezier curve control points, and the maximum thickness of NACA00xx airfoils. It was represented that the use of the Bezier curve significantly improves the optimization performance to reach the optimal shape. The results obtained in test cases presented in this paper show that the proposed sensitivity analysis method reduces the computation cost even for large number of the design variables (Test Case 4) and confirm accuracy and efficiency of the proposed shape optimization algorithm.

Conflict of Interests

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

References

[1] R. M. Hicks, E. M. Murman, and G. N. Vanderplaats, *An Assessment of Airfoil Design by Numerical Optimization*, 1974.

[2] R. M. Hicks and P. A. Henne, "Wing design by numerical optimization," *Journal of Aircraft*, vol. 15, no. 7, pp. 407–412, 1978.

[3] O. Pironneau, "On optimum design in fluid mechanics," *Journal of Fluid Mechanics*, vol. 64, pp. 97–110, 1974.

[4] A. Jameson, "Aerodynamic design via control theory," *Journal of Scientific Computing*, vol. 3, pp. 233–260, 1988.

[5] A. Jameson, "Computational aerodynamics for aircraft design," *Science*, vol. 245, no. 4916, pp. 361–371, 1989.

[6] A. Jameson, "Optimum aerodynamic design using CFD and control theory," AIAA Paper 95-1729, 1995.

[7] A. Jameson and J. Reuther, *Control Theory Based Airfoil Design Using the Euler Equations*, Research Institute for Advanced Computer Science, NASA Ames Research Center, 1994.

[8] M. B. Giles and N. A. Pierce, "Adjoint equations in CFD: duality, boundary conditions and solution behaviour," *AIAA Paper*, vol. 97, p. 1850, 1997.

[9] M. B. Giles and N. A. Pierce, "On the properties of solutions of the adjoint Euler equations," *Numerical Methods for Fluid Dynamics*, pp. 1–16, 1998.

[10] M. B. Giles, *Discrete Adjoint Approximations with Shocks*, Springer, New York, NY, USA, 2003.

[11] M. B. Giles, M. C. Duta, J. Müller, and N. A. Pierce, "Algorithm developments for discrete adjoint methods," *AIAA Journal*, vol. 41, no. 2, pp. 198–205, 2003.

[12] O. Baysal and M. E. Eleshaky, "Aerodynamic design optimization using sensitivity analysis and computational fluid dynamics," *AIAA Journal*, vol. 30, no. 3, pp. 718–725, 1992.

[13] S. Ta'asan, G. Kuruvila, and M. Salas, "Aerodynamic design and optimization in one shot," in *Proceedings of the 30th AIAA Aerospace Sciences Meeting and Exhibit*, Reno, Nev, USA, 1992.

[14] H. Cabuk, C.-H. Sung, and V. Modi, "Adjoint operator approach to shape design for internal incompressible flows," in *Proceedings of the 3rd International Conference on Inverse Design Concepts and Optimization in Engineering Sciences (ICIDES-3 '91)*, pp. 391–404, 1991.

[15] M. Desai and K. Ito, "Optimal controls of Navier-Stokes equations," *SIAM Journal on Control and Optimization*, vol. 32, no. 5, pp. 1428–1446, 1994.

[16] J. Elliott and J. Peraire, "Aerodynamic design using unstructured meshes," AIAA Paper, 1996.

[17] J. Elliott and J. Peraire, "Aerodynamic optimization on unstructured meshes with viscous effects," *AIAA Paper*, vol. 97, p. 1849, 1997.

[18] J. Elliott and J. Peraire, "Practical three-dimensional aerodynamic design and optimization using unstructured meshes," *AIAA Journal*, vol. 35, no. 9, pp. 1479–1485, 1997.

[19] W. K. Anderson and V. Venkatakrishnan, "Aerodynamic design optimization on unstructured grids with a continuous adjoint formulation," *Computers and Fluids*, vol. 28, no. 4-5, pp. 443–480, 1999.

[20] L. Gonzalez, E. Whitney, K. Srinivas, and J. Périaux,, "Multidisciplinary aircraft design and optimisation using a robust evolutionary technique with variable Fidelity models," in *Proceedings of the 10th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference*, vol. 4625, 2004.

[21] I. C. Parmee and A. H. Watson, "Preliminary airframe design using co-evolutionary multiobjective genetic algorithms," in *Proceedings of the Genetic and Evolutionary Computation Conference*, pp. 1657–1665, 1999.

[22] S. Obayashi, "Multidisciplinary design optimization of aircraft wing planform based on evolutionary algorithms," in *Proceedings of the IEEE International Conference on Systems, Man, and Cybernetics*, pp. 3148–3153, October 1998.

[23] A. Oyama, M.-S. Liou, and S. Obayashi, "Transonic axial-flow blade shape optimization using evolutionary algorithm and three-dimensional Navier-Stokes solver," in *Proceedings of the 9th AIAA/ISSMO Symposium and Exhibit on Multidisciplinary Analysis and Optimization*, Atlanta, Ga, USA, 2002.

[24] H.-S. Chung, S. Choi, and J. J. Alonso, "Supersonic business jet design using knowledge-based genetic algorithm with adaptive, unstructured grid methodology," in *Proceedings of the 21st Applied Aerodynamics Conference*, 2003.

[25] A. Jameson and K. Ou, "Optimization methods in computational fluid dynamics," in *Encyclopedia of Aerospace Engineering*, John Wiley & Sons, New York, NY, USA, 2010.

[26] D. Thévenin and G. Janiga, *Optimization and Computational Fluid Dynamics*, Springer, 2008.

[27] B. Mohammadi and O. Pironneau, *Applied Shape Optimization for Fluids*, Oxford University Press, Oxford, UK, 2009.

[28] P. Castonguay and S. K. Nadarajah, "Effect of shape parameterization on aerodynamic shape optimization," in *Proceedings of the 45th AIAA Aerospace Sciences Meeting and Exhibit*, pp. 8–11, January 2007.

[29] F. Mohebbi and M. Sellier, "Optimal shape design in heat transfer based on body-fitted grid generation," *International Journal for Computational Methods in Engineering Science and Mechanics*, vol. 14, no. 3, pp. 227–243, 2013.

[30] F. Mohebbi and M. Sellier, "Three-dimensional optimal shape design in heat transfer based on body-fitted grid generation," *International Journal for Computational Methods in Engineering Science and Mechanics*, vol. 14, no. 6, pp. 473–490, 2013.

[31] J. F. Thompson, F. C. Thames, and C. W. Mastin, "Automatic numerical generation of body-fitted curvilinear coordinate system for field containing any number of arbitrary two-dimensional bodies," *Journal of Computational Physics*, vol. 15, no. 3, pp. 299–319, 1974.

[32] M. W. Kutta, *Lifting Forces in Flowing Fluids*, 1902.

[33] F. Mohebbi and M. Sellier, "On the Kutta condition in potential flow over airfoil," *Journal of Aerodynamics*, vol. 2014, Article ID 676912, 10 pages, 2014.

[34] J. D. Anderson, *Fundamentals of Aerodynamics*, McGraw-Hill, 2001.

[35] J. A. Samareh, "A survey of shape parameterization techniques," in *NASA Conference Publication*, pp. 333–344, Citeseer, 1999.

[36] D. F. Rogers, *An Introduction to NURBS: With Historical Perspective*, Morgan Kaufmann Publications, 2001.

[37] M. Hepperle, "Javafoil—analysis of airfoils," 2008, http://www.mh-aerotools.de/airfoils/javafoil.htm.

[38] A. Jameson, "Aerodynamic shape optimization using the adjoint method," Lectures at the Von Karman Institute, Brussels, Belgium, 2003.

[39] A. Jameson, L. Martinelli, and N. A. Pierce, "Optimum aerodynamic design using the Navier-Stokes equations," *Theoretical and Computational Fluid Dynamics*, vol. 10, no. 1–4, pp. 213–237, 1998.

[40] M. H. Straathof, *Shape parameterization in aircraft design: a Novel method, based on B-splines [Dissertation]*, Delft University of Technology, 2012.

[41] M. B. Giles and N. A. Pierce, "An introduction to the adjoint approach to design," *Flow, Turbulence and Combustion*, vol. 65, no. 3-4, pp. 393–415, 2000.

[42] W. K. Anderson and D. L. Bonhaus, "Airfoil design on unstructured grids for turbulent flows," *AIAA Journal*, vol. 37, no. 2, pp. 185–191, 1999.

[43] T. D. Economon, P. Francisco, and J. J. Alonso, "Optimal shape design for open rotor blades," in *Proceedings of the 30th AIAA Applied Aerodynamics Conference*, pp. 1414–1436, June 2012.

[44] S. K. Nadarajah, *The Discrete Adjoint Approach to Aerodynamic Shape Optimization*, Citeseer, 2003.

Hindawi Publishing Corporation
Mathematical Problems in Engineering
Volume 2014, Article ID 391942, 7 pages
http://dx.doi.org/10.1155/2014/391942

Research Article

An Inversely Designed Model for Calculating Pull-In Limit and Position of Electrostatic Fixed-Fixed Beam Actuators

Cevher Ak and Ali Yildiz

Department of Electrical-Electronics Engineering, Mersin University, 33343, Mersin, Turkey

Correspondence should be addressed to Ali Yildiz; aliyildiz99@gmail.com

Received 11 April 2014; Revised 20 July 2014; Accepted 20 July 2014; Published 20 August 2014

Academic Editor: Fatih Yaman

This study presents an inverse approach to obtain a relation between applied voltage and displacement of the midpoint of fixed-fixed beam actuator. The approach has two main sections. The first one is the inverse design of a model to replace real action of upper beam under electrostatic force. The formula obtained from the first section does not comprise the residual stress and gives very small errors when there is no residual stress on the upper electrode. So, the second part was carried out to add this important system variable into the formula. Likewise, inverse solution was again applied in the later section. The final formula demonstrates that pull-in limit of clamped-clamped actuator is to be at around 40% of original spacing that is in agreement with simulation and previous experimental results. Its percentage errors are within 2% when compared with simulations that are based on finite element method (FEM). The results are comparable to numerical solutions received from diverse distributed models which require more calculation power in electrostatic and structural domains. On top of that, our formula is valid for all displacements from original position up to pull-in limit.

1. Introduction

Electrostatic MEMS based actuators have been used widely as a sensor due to their higher sensitivity, smaller dimensions, low-power consumption, and easy fabrication with new design possibilities. They have been used as a microelectromechanical varactor [1], as a capacitive pressure sensor for measuring blood pressure for a cardiovascular catheter [2], as a mass sensor [3], as an RF microswitch [4, 5], as an energy harvester for MEMS devices [6], and as small force detection [7].

An electrostatically controlled MEMS based fixed-fixed actuator is made up of two parallel conductive beams; the ground electrode is fabricated on a substrate and not movable and the upper electrode is suspended above it with an initial gap (g) and fixed from both ends. When a voltage difference is applied between bottom and upper electrodes, the middle section of top electrode moves towards bottom electrode due to electrostatic force. The counteract spring force will stop the motion of upper electrode at some equilibrium point. The spring force is a linear function of the movement whereas the electrostatic force increases with the square of distance. After a certain point, the restoring force due to bending cannot balance the electrostatic force any longer. The upper electrode is instable after this point and collides on the bottom electrode. This limit is called pull-in limit and voltage value is named as pull-in voltage. The real behavior of upper electrode can be seen in Figure 1. It obeys two constraints; fixed ends have zero movement and zero-angle.

Calculating the pull-in voltage value accurately is the most crucial issue in MEMS actuators. Nevertheless, behavior is nonlinear because of electrostatic and mechanical coupling. Hence, getting an analytical formula for pull-in limit is very difficult. For more than two decades, researchers have been developing many models and methods for electrostatic MEMS based actuators to calculate pull-in limit [8–16]. However, most of these works have found different pull-in limits. When it is controlled by software (COMSOL) which employs finite element method, pull-in limit appears to be at approximately 40% of initial gap which is also consistent

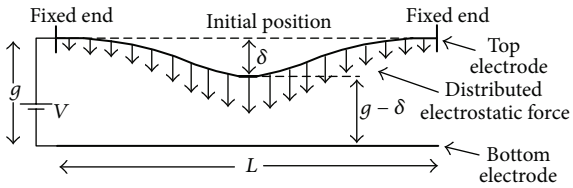


Figure 1: Electrostatic actuator (side view).

Table 1: COMSOL Simulation Pull-in Results of a Fixed-Fixed Beam with length = 200 μm, width = 50 μm, thickness = 1 μm, Young Modulus = 160 GPa, Poisson Ratio = 0.22.

Initial Gap (μm)	Pull-in Gap (μm)	Pull-in Gap/Initial Gap
2	0.797	0.3985
4	1.594	0.3985
5	1.992	0.3984
10	3.984	0.3984

with former experimental measurements [12, 14]. Table 1 points some simulation results out for varying initial gaps. Upper beam was simulated as linear and isotropic poly-Si material in 3D geometry. Its length was chosen as 200 μm for simulation purposes only since length is not a function of pull-in gap. Ratio of pull-in gap to initial gap is constant and independent of material properties of the upper electrode. Other properties of the material in simulation can be seen in Table 1.

The main intention of this work is to find a relatively simple formula which demonstrates not only the pull-in limit at the actual position, but also a relationship between applied potential difference and displacement of the middle part of a clamped-clamped electrostatic actuator.

2. Inversely Designed Model

We have altered the real behavior of upper electrode with a new model for the sake of simplicity. Upper electrode has subdivided into three regions with four pivots. Two of the pivots are located at both fixed ends. They can only rotate around the pivot points whereas they cannot move horizontally and vertically. The other pivot points are located at $3L/7$ distances from both fixed sides symmetrically. These pivots are free to move vertically though. The new model can be seen clearly in Figure 2. Total length of the electrode is L. The first and third partitions divided as $3L/7$ while the second one as $L/7$. These values have been inversely reached by trial and error method since we already know the true results from simulations and previous experimental studies.

We had studied bisection model for fixed-free actuators earlier [21, 22]. The model was dividing the upper electrode into two partitions. One is not moving at all and fixed to the substrate from one end and the other section can freely move linearly around a pivot which is selected by trial and error method inversely. It was successful as well when compared with simulation results and experimental studies. Later, model for fixed-free actuators was simplified further

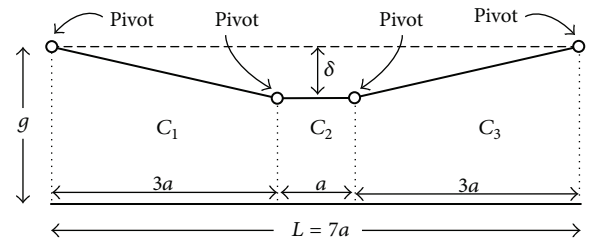


Figure 2: The movements and capacitances of the partitions introduced in new model.

and named as Pivot Model [23]. In the model, fixed side of the upper electrode is pivoted and the other side is totally free to move linearly around the pivot point. The upper beam was assumed as a rigid body in the model for the sake of simplicity. These were reasonable assumptions since the bending of the beam was very small. It establishes a good relation between applied voltage and displacement span from rest position to pull-in limit.

The new model was developed for clamped-clamped actuators and utilizes only one constraint of the fixed ends of top electrode. Fixed ends still have zero displacement while zero-angle constraint is omitted as in Pivot Model. Thus, only one end of the first and third partitions moves linearly around fixed pivot while the second region is totally free to move vertically. The new design model was named as Inverse Pivot Model (IPM) since pivots were used to make the system uncomplicated. In this model, upper electrode is subdivided into three partitions. Locations of the movable pivots were chosen inversely from simulation results and previous experimental measurements. It was also possible to subdivide the beam into more than 3 partitions in order to have a closer representation of the real bending shape of the beam. However, model would get complicated which is not a desired case. In this study, keeping the model as simple as possible is foremost and initiative intention while it gives a good approximation of the real system. After many comparisons with simulation outcomes and previous measurements, length of the upper beam was subdivided into 7 equal segments. One segment was taken as a middle partition, and the other 2 partitions equally shared the rest of the segments due to symmetry. Therefore, side partitions (partitions 1 and 3) consist of 3 segments.

The capacitance calculation of the second partition is easy since it establishes a parallel plate capacitance shape during its movement and C_2 can be calculated as

$$C_2 = \frac{1}{7}\frac{\varepsilon_0 w L}{(g-\delta)}, \tag{1}$$

where ε_0 is permittivity of free space. Capacitances of the first and third partitions are equal because of the symmetry. Therefore, calculating one of them will be enough. However, calculation of it is not simple as second partition.

Partial capacitance of C_1 can be considered as a parallel plate capacitance since the inclination of the upper electrode

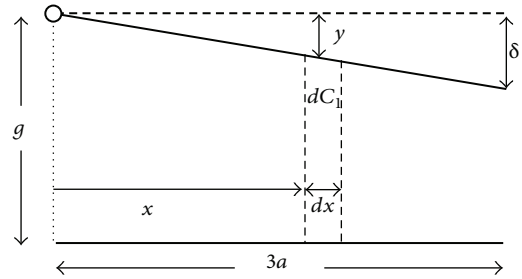


Figure 3: Capacitance calculation for the first partition.

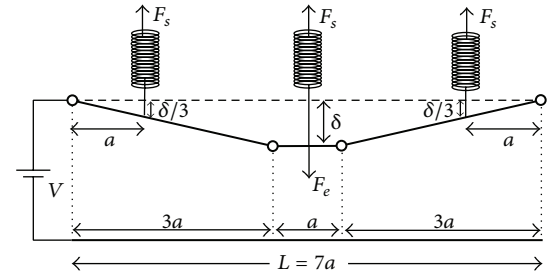


Figure 4: Locating the equivalent forces of new model.

is negligible for this infinitesimally small region. It can be seen in Figure 3 and can be written as

$$dC_1 = \varepsilon_0 \frac{dA}{g - y}. \tag{2}$$

dA is infinitesimal surface area and equal to wdx, where w is the width of the upper beam. y can be found from the geometry shown in Figure 3 as

$$y = \delta \frac{x}{3a}, \tag{3}$$

where δ is the farthest displacement of the upper beam when a potential difference is applied. It equals the displacement of middle point as well. Therefore, (2) can be rewritten as

$$dC_1 = 3a\varepsilon_0 \frac{wdx}{3ag - gx}. \tag{4}$$

So, capacitances of the first partitions can be calculated as

$$C_1 = \int dC_1 = 3a\varepsilon_0 w \int_{x=0}^{3a} \frac{dx}{3ag - gx}. \tag{5}$$

When value of a is inserted in (5), C_1 and C_3 can be calculated as

$$C_1 = C_3 = \frac{3}{7} \frac{\varepsilon_0 wL}{\delta} \ln\left(\frac{g}{g - \delta}\right). \tag{6}$$

In a real actuator, electrostatic force is nonuniform and distributed along the upper beam. It is extremely difficult to express it in a simple formula. Therefore, it has been exchanged with a single equivalent electrostatic force term. Since the middle region is the closest part of the upper electrode to bottom one, the electrostatic force has the biggest component at this point. Besides, because of symmetry, electrostatic force is placed at the center of the upper electrode as a single equivalent force. Nevertheless, fringing effect of the capacitances was ignored in order to keep the model simple. To compensate the absence of the fringing effect, restoring forces are adjusted accordingly. Instead of using one restoring force term at the center, it has been split into three. Two restoring forces are placed at the one-third of the linearly moving partitions from fixed ends in order to decrease effect of spring force term to make up the missing fringing effect. The final exact point is again reached after many attempts by inverse approach. Optimization algorithms can also be used to find a more precise location in the future studies. The middle spring force is apparently placed at the center. All equivalent forces and their locations can be seen in Figure 4.

The infinitesimal electrostatic force term can be written as

$$dF_e = \frac{1}{2} \frac{dC}{d\delta} V^2, \tag{7}$$

where C is the total capacitance term of the system and V is potential difference applied between electrodes. The total electrostatic force term can be calculated as

$$F_e = -\frac{\varepsilon_0 wL}{14} \times \left(\frac{5\delta^2 - 6\delta g + \left(6\delta^2 + 6g^2 - 12\delta g\right) \ln\left(g/\left(g - \delta\right)\right)}{\delta^2 \left(g - \delta\right)^2} \right) V^2 \tag{8}$$

and spring constant k can be obtained for a fixed-fixed beam as [24]

$$k = 32\widehat{E}w\left(\frac{t}{L}\right)^3, \tag{9}$$

where

$$\widehat{E} = \frac{E}{\left(1 - \nu^2\right)}, \tag{10}$$

where E, ν, and t are Young's modulus, Poisson ratio of material, and the thickness of the upper electrode, respectively. Instead of plain E, $\widehat{E}$ has been used to keep the formula valid for wide beams $w \geq 5t$ [17].

Since the 3 restoring springs are parallel to each other, spring constant of each partition is equal to $k/3$. The spring displacement of middle partition is a distance δ, while displacements of the other two partitions are $\delta/3$. The total restoring force term can be written for 3 partitions as

$$F_y = \delta\left(\frac{k}{3}\right) + 2\left(\frac{\delta}{3}\right)\left(\frac{k}{3}\right) = \frac{160}{9} \frac{\delta E w}{\left(1 - \nu^2\right)} \left(\frac{t}{L}\right)^3. \tag{11}$$

Table 2: Comparison of Voltage Values for Arbitrary Displacements. $E = 169$ GPa, $L = 350\,\mu$m, $t = 3\,\mu$m, $\nu = 0.06$ and $g = 1\,\mu$m.

Displacement (μm)	Voltage (Volt) IPM (9)	Voltage (Volt) COMSOL	Δ%
0.0116	4.9518	5	0.97
0.0493	9.9076	10	0.93
0.0739	11.8935	12	0.90
0.1062	13.8822	14	0.85
0.1496	15.8724	16	0.80
0.1775	16.8617	17	0.82
0.2125	17.8567	18	0.80
0.2594	18.8394	19	0.85
0.3445	19.8053	20	0.98
0.3891	19.9634	20.19	1.14

The first term in (11) represents the restoring force of the middle partition spring, and the second term represents the total restoring forces of the side partitions springs.

Since the electrostatic and restoring forces have to be equal to each other at equilibrium position, (8) and (11) can be equated to each other as

$$-\frac{\varepsilon_0 wL}{14}\left(\frac{5\delta^2-6\delta g+\left(6\delta^2+6g^2-12\delta g\right)\ln\left(g/\left(g-\delta\right)\right)}{\delta^2\left(g-\delta\right)^2}\right)\times V^2=\frac{160}{9}\frac{\delta E w}{\left(1-\nu^2\right)}\left(\frac{t}{L}\right)^3. \tag{12}$$

Consequently, the relation between displacement and applied voltage can be obtained from (12) as

$$V=\left(\frac{2240}{9}\left(-Et^3\delta^3(\delta-g)^2\times\left(\varepsilon_0 L^4\left(1-\nu^2\right)\times\left(\ln\left(\frac{g}{g-\delta}\right)\left(6\delta^2+6g^2-12\delta g\right)+5\delta^2-6\delta g\right)\right)^{-1}\right)\right)^{1/2}. \tag{13}$$

Equation (13) is valid not only for pull-in limit, but also for all values within the pull-in limit. A comparison of simulation results received from COMSOL package and our IPM results for different displacements up to pull-in limit boundary. All results seem within 1% error level.

Many fixed-fixed actuator sensors employ only on/off position like a switch. Therefore, determination of pull-in voltage is very crucial for an actuator before real device fabrication in order to produce a proper sensor which is working in a range of interest. In order to retrieve the pull-in limit, derivative of V in (13) with respect to δ has to be taken and equate the result to zero and this position is named as pull-in limit [9]. The upper beam will be unstable and collapse towards bottom beam after this critical point. Since the derivative equation equals zero, we can use only the numerator of the derivative since denominator is not affecting the result. So, denominator is dropped from the derivative equation. Subsequently, some common variables are also canceled in order to keep the calculation simple:

$$18\ln\left(\frac{g}{g-\delta}\right)(g-\delta)^3-\delta\left(2g-3\delta\right)\left(9g-7\delta\right)=0. \tag{14}$$

Unfortunately, having an analytical solution to this equation is very cumbersome. Thence, a computational result has been obtained by iteration as

$$\delta_{\text{Pull-in Limit}}\cong 0.4g. \tag{15}$$

This value also can be seen in Table 2. When δ/g gets closer to 0.4, voltage value starts to saturate. When this important value was inserted into (13) back, crucial value of potential difference can be found as

$$V_{\text{Pull-in Limit}}=\sqrt{11.5486\frac{Eg^3t^3}{\left(1-\nu^2\right)\varepsilon_0 L^4}}. \tag{16}$$

This value is the potential difference value at the pull-in limit boundary. Applying a voltage difference higher than this value causes the upper electrode to be unstable and collapse onto the bottom electrode.

Table 3: Comparison of IPM and IPMF with previous simulation results. $L = 250\,\mu$m, $E = 169$ GPa, $\nu = 0.06$, $w = 50\,\mu$m, $t = 3\,\mu$m and $g = 1\,\mu$m in each case.

Residual Stress, σ, (MPa)	0	100	−25
CoSolve [17]	40.10	57.60	33.60
V (2D) [17] (Δ%)	39.50 (1.50%)	56.90 (1.22%)	33.70 (0.30%)
OCI Model [12] (Δ%)	39.60 (1.25%)	57.40 (0.35%)	33.71 (0.33%)
GDQM [15] (Δ%)	39.13 (2.41%)	57.62 (0.03%)	33.63 (0.09%)
IPM (Δ%)	39.13 (2.41%)	39.13 (32.07%)	39.13 (16.461%)
IPMF (Δ%)	39.42 (1.70%)	57.03 (0.99%)	33.61 (0.03%)

3. Inverse Pivot Modified Formula

When it is checked with simulation results obtained by a different software package which utilize finite element method from a previous study, it can be seen that formula gives small errors if there is no residual stress. However, error goes as high as 45% whenever residual stress gets involved since our Inverse Pivot Model does not take residual stress into account. Effect of residual stress can be seen in Table 3. Thence, we decided to improve the formula by considering residual stress as a system parameter. In this second section, formula is merely modified inversely since we have a starting point (equation (16)) and the true results from previous

Table 4: Comparison of IPM and IPMF with previous simulation results. $L = 350\,\mu$m, $E = 169$ GPa, $\nu = 0.06$, $w = 50\,\mu$m, $t = 3\,\mu$m and $g = 1\,\mu$m in each case.

Residual Stress, σ, (MPa)	0	100	−25
CoSolve [17]	20.30	35.80	13.70
V (2D) [17] (Δ%)	20.20 (0.50%)	35.40 (1.11%)	13.80 (0.73%)
OCI Model [12] (Δ%)	20.20 (0.50%)	35.91 (0.31%)	13.71 (0.07%)
GDQM [15] (Δ%)	20.36 (0.30%)	35.99 (0.53%)	13.60 (0.73%)
IPM (Δ%)	19.96 (1.67%)	19.96 (44.25%)	19.96 (45.69%)
IPMF (Δ%)	20.11 (0.94%)	35.25 (1.54%)	13.97 (1.97%)

researches [12, 15, 17] to compare outcomes of the new formula. Adding residual stress into (16) is easier than starting from scratch to get a new model. From Tables 3 and 4, it can be apparently seen that positive residual stress increases whereas negative one decreases pull-in voltage. Nevertheless, effect of residual stress should increase with the thickness and decrease with the length of upper electrode. We had to move some physical and system parameters from numerator to denominator in order to keep formula dimensionally correct. After many trials, Inverse Pivot Modified Formula (IPMF) has been reached as

$$V = \left(\frac{\left(\sigma t (t/L)^{1.1} + (65/9)\left(Egt^3/\left(1-\nu^2\right)L^3\right)\right)}{(61/99)\left(\varepsilon_0 L/g^2\right)} \right)^{1/2}, \quad (17)$$

where σ is the residual stress. Equation (17) has not been attained as an analytical calculation. However, when the residual stress is assumed as zero in this equation, it closely approaches (16). It also establishes very good results for actuators with residual stress at pull-in limit boundary. Errors are now in acceptable ranges which all are smaller than 2%, even for the worst case (see Tables 3 and 4).

In Tables 3 and 4, Poisson ratios are 0.06 in all cases. So, our model was also checked with a different Poisson ratio which is 0.32. From Table 5, it can be seen that IPM delivers a comparable result with other studies. In particular, IPMF demonstrates very similar error levels when compared with previous methods which depends on numerical distributed models. These kinds of model require more computing power and need more time to get the result. However, our formula is just one step calculation if it is written in an Excel sheet.

The IPMF is also checked for some other parameters in order to explore the applicable range of the formula. Table 6 shows simulation results obtained in COMSOL for a poly-Si beam whose material properties are selected from COMSOL's library. Mesh numbers are selected automatically for each case in the software.

Table 6 demonstrates comparison between IPMF and simulation results for wide range of material parameters. IPMF gives very satisfactory results with a maximum error level of 8% for residual stress case. In Table 7, width is changed for 200 μm and 500 μm beam lengths while other parameters are kept constant to show the effect of width. IPMF again shows very satisfactory results for the range of width with a maximum error level of 3.5%. In Table 8, initial gap is changed for again 200 μm and 500 μm beam lengths while other parameters are kept constant to show the effect of the gap between electrodes. It can be seen that error level gets increase as the initial gap heightens for both 200 μm and 500 μm beam lengths. This error stems from ignorance of fringing effect between upper and bottom electrodes. As the initial gap rises, the capacitance value of the system also proliferates due to additional increase of the fringing effect.

Table 5: Comparison of IPM and IPMF with previous simulation results [12]. $L = 250\,\mu$m, $E = 169$ GPa, $\nu = 0.32$, $w = 50\,\mu$m, $t = 3\,\mu$m and $g = 1\,\mu$m in each case.

Residual Stress, σ, (MPa)	0
CoSolve FEA [17]	41.20
V (2D) [17] (Δ%)	41.50 (0.73%)
[18] (Δ%)	42.54 (3.25%)
[19] (Δ%)	41.20 (0.00%)
[20] (Δ%)	41.42 (0.53%)
OCI Model [12]	41.72 (1.26%)
IPM (Δ%)	41.53 (0.80%)
IPMF (Δ%)	41.23 (0.07%)

Table 6: Pull-in Voltage Comparison of IPMF to simulation results for Poly-Si material from COMSOL library. $E = 160$ GPa, $\nu = 0.22$ and $w = 50\,\mu$m for all cases.

Residual Stress, σ, (MPa)	COMSOL (V)	IPMF (V)	(Δ%)
$L = 100\,\mu$m, $t = 2\,\mu$m, $g = 1\,\mu$m, Number of Meshes = 7533			
0	134.70	132.53	1.61
−25	130.30	127.74	1.96
100	150.00	150.18	0.12
$L = 200\,\mu$m, $t = 2\,\mu$m, $g = 1\,\mu$m, Number of Meshes = 4467			
0	33.60	33.13	1.40
−25	29.00	28.40	2.07
100	47.50	47.55	0.11
$L = 300\,\mu$m, $t = 3\,\mu$m, $g = 2\,\mu$m, Number of Meshes = 3531			
0	78.30	76.52	2.27
−25	67.90	68.49	0.87
100	109.00	102.51	5.95
$L = 400\,\mu$m, $t = 3\,\mu$m, $g = 2\,\mu$m, Number of Meshes = 2817			
0	44.01	43.04	2.20
−25	34.00	34.88	2.59
100	72.00	66.30	7.92
$L = 500\,\mu$m, $t = 4\,\mu$m, $g = 2\,\mu$m, Number of Meshes = 3259			
0	42.83	42.41	0.98
−25	33.30	32.71	1.77
100	68.10	68.64	0.79
$L = 750\,\mu$m, $t = 5\,\mu$m, $g = 3\,\mu$m, Number of Meshes = 1782			
0	49.10	48.39	1.45
−25	32.80	34.99	6.68
100	87.00	82.53	5.14

Table 7: Comparison of IPMF to simulation results for the variety of widths. $E = 160$ GPa, $\nu = 0.22$ and $w = 50\,\mu$m for all cases.

Width, w, (μm)	COMSOL (V)	IPMF (V)	(Δ%)
$L = 200\,\mu$m, $t = 2\,\mu$m, $g = 1\,\mu$m			
50	33.60	33.13	1.40
20	33.30	33.13	0.51
10	33.10	33.13	0.09
5	32.90	33.13	0.70
2	32.60	33.13	1.63
1	32.20	33.13	2.89
$L = 500\,\mu$m, $t = 4\,\mu$m, $g = 2\,\mu$m			
50	42.83	42.41	0.98
20	42.30	42.41	0.26
10	42.10	42.41	0.74
5	41.80	42.41	1.46
2	41.40	42.41	2.44
1	41.00	42.41	3.44

Table 8: Comparison of IPMF to simulation results for the variety of initial gaps. $E = 160$ GPa, $\nu = 0.22$ and $w = 50\,\mu$m for all cases.

Gap, g, (μm)	COMSOL (V)	IPMF (V)	(Δ%)
$L = 200\,\mu$m, $w = 50\,\mu$m, $t = 2\,\mu$m			
1	33.60	33.13	1.40
2	99.50	93.71	5.82
3	196.00	172.17	12.16
5	520.00	370.45	28.76
$L = 500\,\mu$m, $w = 50\,\mu$m, $t = 4\,\mu$m			
2	42.83	42.41	0.98
3	79.80	77.91	2.37
4	126.00	119.95	4.81
6	249.00	220.38	11.49

4. Conclusions

In the present study, we have proposed a new inversely designed model (IPM) and modified formula (IPMF) which both deliver pull-in limit of a fixed-fixed beam actuator at around 40% of the initial gap. They are consistent with COMSOL simulation results (Tables 1 and 2) and previous experimental and distributed models (Tables 3, 4, and 5). Besides, it establishes a good relation between applied voltage and displacements commencing from rest position up to pull-in limit. Particularly, our inversely designed modified formula yields percentage errors less than 2%, even for the worst case. IPMF is a plain formula which does not demand much computing power. Moreover, IPM is successful with gratifying outcomes for applied voltages for given displacements. When compared with previous experimental measurements, IPM and IPMF can be found considerably successful too. However, if the upper electrode has residual stress, our first model IPM cannot deliver good results because model does not take residual stress into account. On the other hand, IPMF still demonstrates very small percentage errors which are comparable to previous distributed models. IPMF also attests successful outcomes for a wide range of top beam geometric parameters (Tables 6 and 7). The formula is valid for both long beams where $L \geq 5w$ and narrow beams ($w > 5t$) [12].

The most eminent benefit of IPMF is that it establishes a good approximation of the system with an unsophisticated plain formula. One can simply calculate required voltage for pull-in limit rather than utilize numerical distributed methods which requires more computing power and is time consuming.

Although IPM and IPMF are outstandingly precise especially around pull-in limit, they have some limitations since they neglect some physical parameters of the system for sake of simplification of the final formula. Firstly, both of them do not take the fringing effect into account. It causes the actual capacitance to be higher than what we have calculated in our Inverse Pivot Model. Therefore, when the initial gap gets higher, IPMF also deviates from acceptable error levels (Table 8).

Secondly, the upper beam was also assumed as a rigid body in the model. Thence, length of the beam was taken as constant even under significant electrostatic force. Therefore, both axial and transverse stresses would not be formed.

Lastly, the models assume the actuators are in vacuum since they ignore any atmospheric pressure on the upper electrode.

IPMF can be improved more by applying artificial optimization techniques and fringing effect of the capacitance geometry can be taken into account to make the formula deliver smaller error level.

Conflict of Interests

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

Acknowledgment

The authors would like to thank the anonymous reviewers who have helped to improve the quality of their paper.

References

[1] A. L. Roy, A. Bhattacharya, R. R. Chaudhuri, and T. K. Bhattacharyya, "Analysis of the pull-in phenomenon in microelectromechanical varactors," in *Proceedings of the 25th International Conference on VLSI Design*, pp. 185–190, Hyderabad, India, January 2012.

[2] H.-L. Chau and K. D. Wise, "Ultraminiature solid-state pressure sensor for a cardiovascular catheter," *IEEE Transactions on Electron Devices*, vol. 35, no. 12, pp. 2355–2362, 1988.

[3] W. Zhang, R. Baskaran, and K. L. Turner, "Effect of cubic nonlinearity on auto-parametrically amplified resonant MEMS mass sensor," *Sensors and Actuators A*, vol. 102, no. 1-2, pp. 139–150, 2002.

[4] S. P. Pacheco, L. P. B. Katehi, and C. T.-C. Nguyen, "Design of low actuation voltage RF MEMS switch," in *Proceedings of the IEEE MTT-S International Microwave Symposium Digest*, pp. 165–168, Boston, Mass, USA, June 2000.

[5] H.-H. Yang, O. L. Jeong, and J.-B. Yoon, "Maneuvering pull-in voltage of an electrostatic micro-switch by introducing a pre-charged electrode," in *Proceedings of the IEEE International Electron Devices Meeting (IEDM '07)*, pp. 439–442, Washington, DC, USA, December 2007.

[6] D. Shen, J. Park, J. Ajitsaria, S. Choe, H. C. Wikle, and D. Kim, "The design, fabrication and evaluation of a MEMS PZT cantilever with an integrated Si proof mass for vibration energy harvesting," *Journal of Micromechanics and Microengineering*, vol. 18, no. 5, Article ID 055017, 2008.

[7] P. Mohanty, D. A. Harrington, and M. L. Roukes, "Measurement of small forces in micron-sized resonators," *Physica B: Condensed Matter*, vol. 284–288, part 2, pp. 2143–2144, 2000.

[8] M. Mojahedi, M. Moghimi Zand, and M. T. Ahmadian, "Static pull-in analysis of electrostatically actuated microbeams using homotopy perturbation method," *Applied Mathematical Modelling*, vol. 34, no. 4, pp. 1032–1041, 2010.

[9] G. N. Nielson and G. Barbastathis, "Dynamic pull-in of parallel-plate and torsional electrostatic MEMS actuators," *Journal of Microelectromechanical Systems*, vol. 15, no. 4, pp. 811–821, 2006.

[10] Y. Hu and G. Lee, "A closed form solution for the pull-in voltage of the micro bridge," *Tamkang Journal of Science and Engineering*, vol. 10, no. 2, pp. 147–150, 2007.

[11] L. Mol, E. Cretu, L. A. Rocha, and R. F. Wolffenbuttel, "Full-gap positioning of parallel-plate electrostatic MEMS using on-off control," in *Proceedings of the IEEE International Symposium on Industrial Electronics (ISIE '07)*, pp. 1464–1468, Vigo, Spain, June 2007.

[12] S. Chowdhury, M. Ahmadi, and W. C. Miller, "Pull-in voltage study of electrostatically actuated fixed-fixed beams using a VLSI on-chip interconnect capacitance model," *Journal of Microelectromechanical Systems*, vol. 15, no. 3, pp. 639–651, 2006.

[13] K. O. Owusu and F. L. Lewis, "Solving the "pull-in" instability problem of electrostatic microactuators using nonlinear control techniques," in *Proceedings of the 2nd IEEE International Conference on Nano/Micro Engineered and Molecular Systems (IEEE NEMS '07)*, pp. 1190–1195, Bangkok, Thailand, January 2007.

[14] Y. C. Hu, C. M. Chang, and S. C. Huang, "Some design considerations on the electrostatically actuated microstructures," *Sensors and Actuators A: Physical*, vol. 112, no. 1, pp. 155–161, 2004.

[15] H. Sadeghian, G. Rezazadeh, and P. M. Osterberg, "Application of the generalized differential quadrature method to the study of pull-in phenomena of MEMS switches," *Journal of Microelectromechanical Systems*, vol. 16, no. 6, pp. 1334–1340, 2007.

[16] H. C. Nathanson, W. E. Newell, R. A. Wickstrom, and J. R. Davis Jr., "The resonant gate transistor," *IEEE Transaction of Electron Devices*, vol. 14, no. 3, pp. 117–133, 1967.

[17] P. M. Osterberg and S. D. Senturia, "M-test: a test chip for MEMS material property measurement using electrostatically actuated test structures," *Journal of Microelectromechanical Systems*, vol. 6, no. 2, pp. 107–118, 1997.

[18] S. Pamidighantam, R. Puers, K. Baert, and H. A. C. Tilmans, "Pull-in voltage analysis of electrostatically actuated beam structures with fixed-fixed and fixed-free end conditions," *Journal of Micromechanics and Microengineering*, vol. 12, no. 4, pp. 458–464, 2002.

[19] C. O'Mahony, M. Hill, R. Duane, and A. Mathewson, "Analysis of electromechanical boundary effects on the pull-in of micromachined fixed-fixed beams," *Journal of Micromechanics and Microengineering*, vol. 13, no. 4, pp. S75–S80, 2003.

[20] H. A. C. Tilmans and R. Legtenberg, "Electrostatically driven vacuum-encapsulated polysilicon resonators. Part II. Theory and performance," *Sensors and Actuators A*, vol. 45, no. 1, pp. 67–84, 1994.

[21] A. Yildiz, C. Ak, and H. Canbolat, "New approach to pull-in limit and position control," in *Electrostatics*, chapter 6, pp. 139–150, In-Teh, Rijeka, Croatia, 2012.

[22] C. Ak, *Dynamic position control of electrostatic actuators [M.S. thesis]*, Fen Bilimleri Enstitusu, Mersin University, Mersin, Turkey, 2008.

[23] C. Ak and A. Yildiz, "Development of a novel analytical model for calculating pull-in limit and voltage value for a desired position of electrostatic cantilever free tip," *Pensee Journal*, vol. 76, pp. 360–373, 2014.

[24] S. C. Saha, U. Hanke, G. U. Jensen, and T. Saether, "Modeling of spring constant and pull-down voltage of non uniform RF MEMS cantilever," in *Proceedings of the IEEE International Behavioral Modeling and Simulation Workshop*, pp. 56–60, September 2006.

Hindawi Publishing Corporation
Mathematical Problems in Engineering
Volume 2014, Article ID 369369, 13 pages
http://dx.doi.org/10.1155/2014/369369

Research Article

Applying Hybrid Heuristic Approach to Identify Contaminant Source Information in Transient Groundwater Flow Systems

Hund-Der Yeh, Chao-Chih Lin, and Bo-Jei Yang

Institute of Environmental Engineering, National Chiao Tung University, 1001 University Road, Hsinchu 30010, Taiwan

Correspondence should be addressed to Hund-Der Yeh; hdyeh@mail.nctu.edu.tw

Received 9 April 2014; Accepted 25 July 2014; Published 18 August 2014

Academic Editor: Tzu-Yang Yu

Simultaneous identification of the source location and release history in aquifers is complicated and time-consuming if the release of groundwater contaminant source varies in time. This paper presents an approach called SATSO-GWT to solve complicated source release problems which contain the unknowns of three location coordinates and several irregular release periods and concentrations. The SATSO-GWT combines with ordinal optimization algorithm (OOA), roulette wheel approach, and a source identification algorithm called SATS-GWT. The SATS-GWT was developed based on simulated annealing, tabu search, and three-dimensional groundwater flow and solute transport model MD2K-GWT. The OOA and roulette wheel method are utilized mainly to reduce the size of feasible solution domain and accelerate the identification of the source information. A hypothetic site with one contaminant source location and two release periods is designed to assess the applicability of the present approach. The results indicate that the performance of SATSO-GWT is superior to that of SATS-GWT. In addition, the present approach works very effectively in dealing with the cases which have different initial guesses of source location and measurement errors in the monitoring points as well as problems with large suspicious areas and several source release periods and concentrations.

1. Introduction

The issues of identifying the source location and/or recovering the release history of a groundwater contaminant plume are getting more and more public concern recently. In some countries, groundwater is an important source for drinking water and agricultural use. If a site is found to have groundwater contamination, the source information including the location as well as release concentration and period should be determined before taking remedial actions. Site remediation is usually quite costly, so the responsible parties for the contamination should be recognized via the source identification works. In addition, incorrect information on contaminant source may confuse or mislead remediation plan. The technique for identifying contaminant source location and its release history is therefore important in dealing with the groundwater contamination problem. Moreover, if the source release varies in time, the estimation of the actual source information becomes rather complicated and difficult. Thus, there is a need to develop an effective approach for identifying the contaminant source location and its release history based on the concentration measurements.

Identification of unknown contaminant sources in groundwater is an inverse problem. Mathematically, the processes of contaminant transport in groundwater are irreversible and their inversion solutions are sensitive to the errors in the observation data, especially when data are sparse or missing (e.g., [1–6]). Atmadja and Bagtzoglou [1] pointed out that the groundwater source identification is an ill-posed problem because the solution may not be unique and stable. They reviewed the available methods for source location identification and the release history reconstruction and classified them into four categories: optimization approaches, probabilistic and geostatistical (GS) simulation approaches, analytical solution and regression approaches, and direct approaches. Tracking the contaminant source location usually needs to run forward simulations with an initial guess solution and then to search the best-fitted solution via an optimization approach. Probabilistic and GS simulation approaches employ several probabilistic and statistical techniques to assess the probability of the location of the sources (Sun [7]). Atmadja and Bagtzoglou [1] indicated that this approach is applicable only when

the location of the potential source is known in advance. Regression approaches along with analytical solution can estimate all the parameters simultaneously but work well only for simple aquifer geometry and flow conditions. Direct approaches reconstruct the release history by solving governing equation directly.

The problems of identifying groundwater contaminant source can generally be classified as (1) identifying source location problems (e.g., [8–17]), (2) recovering the release history problems (e.g., [5, 18–34]), and (3) identifying source location and recovering the release history problems simultaneously (e.g., [2, 7, 35–42]).

For the third-type problem, Aral and Gaun [35] proposed an approach called improved genetic algorithm (IGA) to determine the information of source location, leak rate, and release period. They showed that the results obtained from the IGA match with those obtained from linear and nonlinear programming approaches. Later, they further developed an iterative approach based on genetic algorithm (GA) algorithm, defined as progressive genetic algorithm (PGA), in conjunction with a finite element groundwater flow model (Aral et al. [43]) to identify the coordinates of source location and release history in a two-dimensional steady-state groundwater flow system. In their case study, the source release history was assumed to be over 180 months and four monitor wells were installed at the downstream of the source location. The contaminant concentration was sampled at each well every 3 months. Therefore, there were totally 240 observation data used in identifying the source location and reconstructing the release history. Mahar and Datta [2] used an optimal source identification methodology based on embedded optimization models to estimate the release concentrations of multiple hypothetical contaminant sources over discrete time intervals using breakthrough curve data. Their study successfully identified the source information for flow in both steady and transient states. Bagtzoglou [36] modified the reversible-time particle tracking method by introducing a variance minimization procedure to backtrack groundwater solute concentration profiles and identify the most probable source location. Bagtzoglou and Atmadja [37] presented a comprehensive literature review on the mathematical methods for hydrologic inversion and the identifications of the contaminant source location and time-release history. Neupauer and Lin [38] extended the work of Neupauer and Wilson [44] by conditioning the backward probability density functions of source location to measured concentration data. Sun et al. [39] employed a constrained robust least squares (CRLS) method to recover the release history of a single source, and the results of CRLS in their hypothetic example are better than several classic methods (i.e., ordinary least squares (LS), standard total least squares (TLS), and nonnegative least squares (NNLS)). In addition, they further employed the CRLS combined with a branch-and-bound global optimization solver for identifying source locations and release histories (Sun et al. [40]). In their numerical examples, a two-dimensional and steady-state flow field was developed. Totally 57 sampled observation data from eleven observation wells were utilized for release history identification. Later, Sun [7] developed a robust version of the GS approach, namely, the robust geostatistical (RGS) approach, to explicitly illustrate the contaminant release history identification in a 2D heterogeneous aquifer with the hydraulic conductivity exhibiting spatially lognormal distribution. Yeh et al. [15] constructed a source identification model, SATS-GWT, by combining simulated annealing (SA), tabu search (TS), and MF2K-GWT, to identify the constant source release problem. Their model can determine the information of contaminant source with a constant release rate in a three-dimensional (3D) transient groundwater flow system. Ababou et al. [41] presented a new and stable methodology for pollutant source identification in terms of unknown initial position and past history based on the reverse antidiffusive random walk scheme. Butera et al. [42] introduced the simultaneous release function and source location identification (SRSI), which was capable of simultaneously identifying the source location and release history of the contaminant in 2D confined aquifers with strongly nonuniform flow fields.

Ho et al. [45] presented an approach called ordinal optimization algorithm (OOA) which can solve complex optimization problems, which usually require huge amount of computing time in obtaining the optimal solution, effectively and accurately. The OOA is suitable for solving optimization problems with sifting the most possible solution for further evaluations (e.g., [46–49]).

This study aims at developing a novel approach called SATSO-GWT to identify the source location and release history in a 3D, heterogeneous, and transient groundwater flow system. The approach combines the model MD2K-GWT for simulating the groundwater flow and pollutant transport with heuristic optimization techniques such as SA, TS, OOA, and roulette wheel method. This new approach has the advantages of avoiding possible trap in a local optimum and improving the computational efficiency when the searching space of problem becomes very large. A hypothetic case is design to assess the applicability of the SATSO-GWT. In addition, three cases are considered to assess the performance of SATSO-GWT. They are (1) different initial guesses of source location, (2) various measurement errors in the monitoring points, (3) a large suspicious area with six release periods and concentrations.

2. Methodology

2.1. Groundwater Flow and Transport Simulation. Darcy's law can be written as (Konikow et al. [50])

$$V_i = -\frac{K_{ij}}{\varepsilon}\frac{\partial h}{\partial x_i}, \quad i, j = 1, 2, 3, \tag{1}$$

where V_i is a vector of the average linear velocity of groundwater flow [L/T], ε is the effective porosity (dimensionless), K_{ij} is the hydraulic conductivity tensor of the porous media [L/T], h is the hydraulic head [L], and x_i are the Cartesian coordinates. Combining Darcy's law with the continuity equation, the 3D groundwater flow equation can be expressed as (Konikow et al. [50])

$$\frac{\partial}{\partial x_i}\left(K_{ij}\frac{\partial h}{\partial x_i}\right) = S_s\frac{\partial h}{\partial t} + W, \quad i, j = 1, 2, 3, \tag{2}$$

where S_s is the specific storage [L^{-1}], t is time [T], and W is the volumetric flux per unit volume (positive for inflow and negative for outflow [1/T]). Equation (2) can be used to predict the hydraulic head distribution in a 3D groundwater flow system.

The governing equation for 3D solute transport in groundwater can be written as (Konikow et al. [50])

$$\frac{\partial(\varepsilon C)}{\partial t} + \frac{\partial}{\partial x_i}(\varepsilon C V_i) - \frac{\partial}{\partial x_i}\left(\varepsilon D_{ij}\frac{\partial C}{\partial x_j}\right) - \sum C' W = 0, \quad i, j = 1, 2, 3, \tag{3}$$

where C is the contaminant concentration [M/L^3], D_{ij} is a second-order tensor of the dispersion coefficient [L^2/T], and C' is the concentration of the source or sink fluid [M/L^3]. The average linear velocity V_i can be determined by (1).

The computer model MF2K-GWT developed based on (2) and (3) by the United State Geological Survey can be used to simulate the groundwater flow and contaminant transport simultaneously. This model combines the modular 3D finite-difference ground-water flow model, MODFLOW-2000 (Harbaugh et al. [51]), and the 3D method-of-characteristics solute-transport model (MOC3D) (Konikow et al. [50]) to simulate groundwater flow field and spatial and temporal plume distribution.

2.2. Simulated Annealing. Press et al. [52] mentioned that SA is a technique suitable for solving large-scale optimization problems. The concept of SA is based on an analogy to crystallization of a solid annealing from a high temperature state. If the temperature is cooled properly, a most stable crystalline structure of the solid will be obtained with minimum energy state. The possible solution spaces for a problem to be solved looks like different crystalline structures and the optimal solution of the problem is equivalent to the most stable crystalline structure.

In the SA, the Metropolis mechanism (Metropolis et al. [53]) is employed to determine the acceptance of adjacent solution. This mechanism dictates that the SA is capable of accepting bad trial solution to avoid the problem of being trapped in the local optimal solution. More details of the introduction of SA are available in Metropolis et al. [53] or Yeh et al. [15]. The SA has been successfully applied to various types of problems such as aquifer parameter estimation (e.g., [54–56]), pipe wall surface reaction rate (e.g., [57]), and pumping source information (e.g., [58]).

2.3. Tabu Search. Glover [59] proposed two main concepts of TS: memory and learning. Through memory and learning, the TS is of more intensification and diversification in algorithm. Memory means to memorize the past solutions to avoid the repetition of evaluations. During the process of learning, the result of next experiment infers from the memorized prior result. A better result may encourage the next trial to increase the accuracy of the obtained solution. Then through the learning process, the succeeding search can focus on better solutions but not wasting time on worse solutions. According to these two ideas, TS utilizes the tabu list and aspiration criterion to interdict or to encourage some trial solutions during the iterative process. The utility of the tabu list is to memorize some previously evaluated trial solutions. The goal of the aspiration criteria is to release some of the solutions memorized in the tabu list to avoid the iteration cycling and may finally be trapped in a local optimal solution.

The TS has been successfully applied to solve groundwater problems such as the identification of optimal parameter structure (Zheng and Wang [60]) and the determination of spatial pattern of groundwater pumping rates (Tung and Chou [61]). The iterative procedure of TS in Yeh et al. [15], which contains the components of initial guess, candidate solution and movement, tabu list, and aspiration criterion, is adopted in this study.

2.4. Ordinal Optimization. Recently, the OOA has been applied to many areas related to simulation-based complex optimization problems. The OOA has two major tenets: ordinal comparison and goal softening procedures. The first procedure is to see if there is a relative relationship between each solution because it is much easier to find better solutions. The second procedure is to determine a reliable and good enough solution instead of directly evaluating the optimal solution in a complex optimization model. Therefore, this procedure reduces the consumption of computation and obtains the optimum solution from the feasible solution space. To get top proportion solutions is much easier than to find out the best one. Lau and Ho [47] showed that the OOA ensures that top 5% solutions can be regarded as good enough solutions and are of very high probability ($\geqq 0.95$) to be reliable.

According to the OOA, all the possible trials are estimated roughly and ranked quickly. The feasible solution domain is divided into several different parts, and the possible optimum solution located in subdomain might be effortlessly recognized. The optimum solution can then be obtained while all the calculation efforts are focused on searching the possible subdomain. A crude model should first be employed to estimate and rank the solution, and the good solutions can be separated from the bad ones. The goal softening procedure then concentrates on the top proportion solutions in order to find the optimum solution. Accordingly, the simulation time can be considerably reduced. The OOA has been successfully applied to many areas such as power system planning and operation (Guan et al. [62]; Lin et al. [63]), electricity network planning (Liu et al. [64]), and wafer testing (Lin and Horng [65]).

2.5. Roulette Wheel Method. The roulette wheel method is an important part of GA. The concept of GA is based on the survival of the fittest by natural selection. Better solutions have larger areas occupied on roulette wheel and the corresponding solutions will have higher chance to be selected. Through the procedure of not evaluating the bad solutions, computer time can be considerably reduced.

2.6. SATSO-GWT Model. A new model called SATSO-GWT is developed based on SATS-GWT, OOA, and the roulette

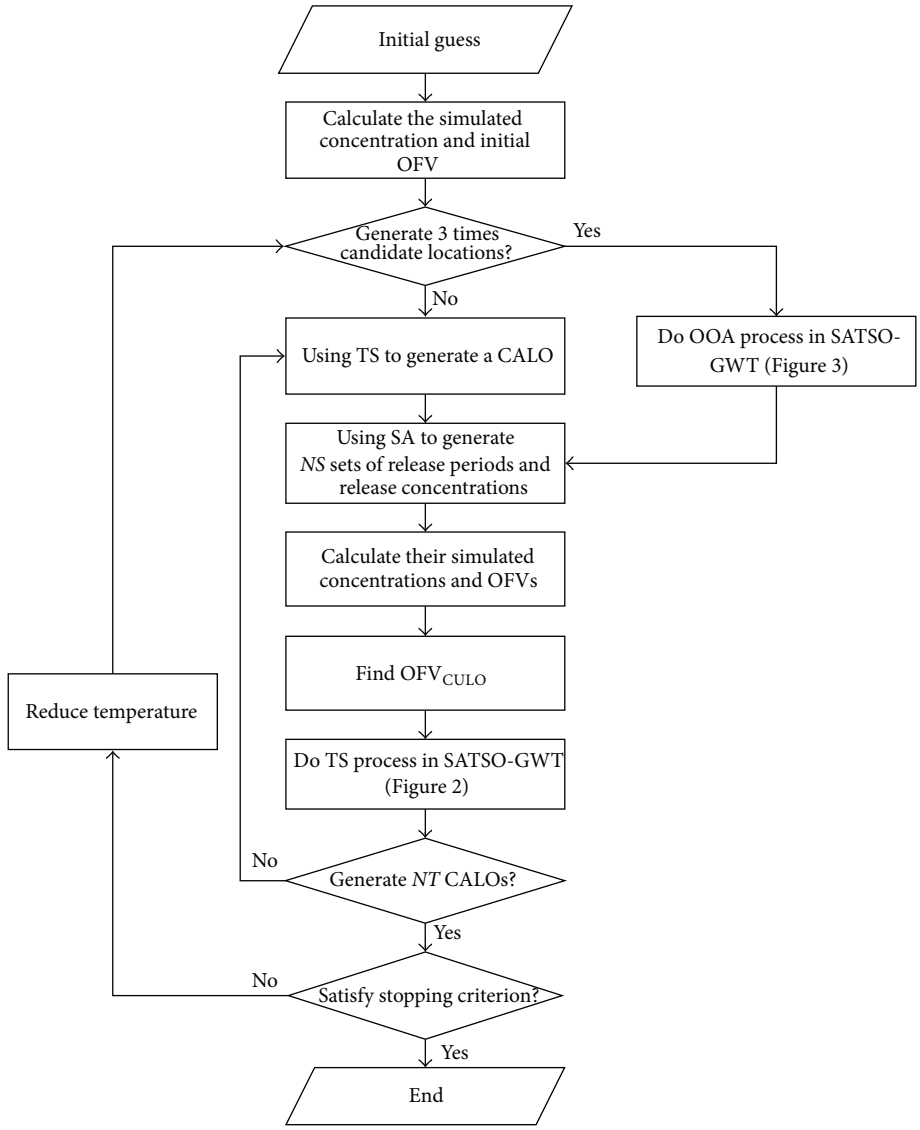


Figure 1: Flowchart of SATSO-GWT. The OFV represents the objective function value, CALO represents the candidate location, and $\mathrm{OFV}_{\mathrm{CULO}}$ represents the optimal OFV at current location.

wheel selection method. The objective function in SATSO-GWT is to minimize the sum of square errors between the simulated and observed concentrations and defined as

$$\text{Minimize } f = \frac{1}{nm \times np} \sum_{j=1}^{np} \sum_{i=1}^{nm} \left(C_{ij,\mathrm{sim}} - C_{ij,\mathrm{obs}}\right)^2, \quad (4)$$

where nm is the total number of monitoring wells, np is the number of observed concentration measured in a monitoring well, $C_{ij,\mathrm{sim}}$ is the simulated concentration at jth terminated time period in ith monitoring well, and $C_{ij,\mathrm{obs}}$ is the observed concentration sampled at jth terminated time period in ith monitoring well. The value $nm \times np$ is generally greater than the number of unknowns (Yeh et al. [15]). Equation (4) is used to calculate the objective function value (OFV) of the trial solution generated by the present approach.

To use MF2K-GWT, the problem domain has to be discretized into block-centered finite difference meshes. A block including several finite difference meshes is then chosen as a suspicious area which includes the contaminant source. Figure 1 displays the flowchart of SATSO-GWT while Figures 2 and 3 show the flowcharts of TS process and OOA in SATSO-GWT, respectively. All meshes in the suspicious area are called candidate source locations (CALOs). The first step of SATSO-GWT is to calculate the initial OFV based on the initial guesses of the source location, release periods, and release concentrations. The initial guess of source location is considered as the current location (CULO) and the initial OFV is set as the current global optimal objective function value ($\mathrm{OFV}_{\mathrm{GO}}$). Then SATSO-GWT generates one CALO and NS trial solutions for the source release periods and concentrations. For each set of trial solutions, MF2K-GWT is employed to predict the simulated concentrations at

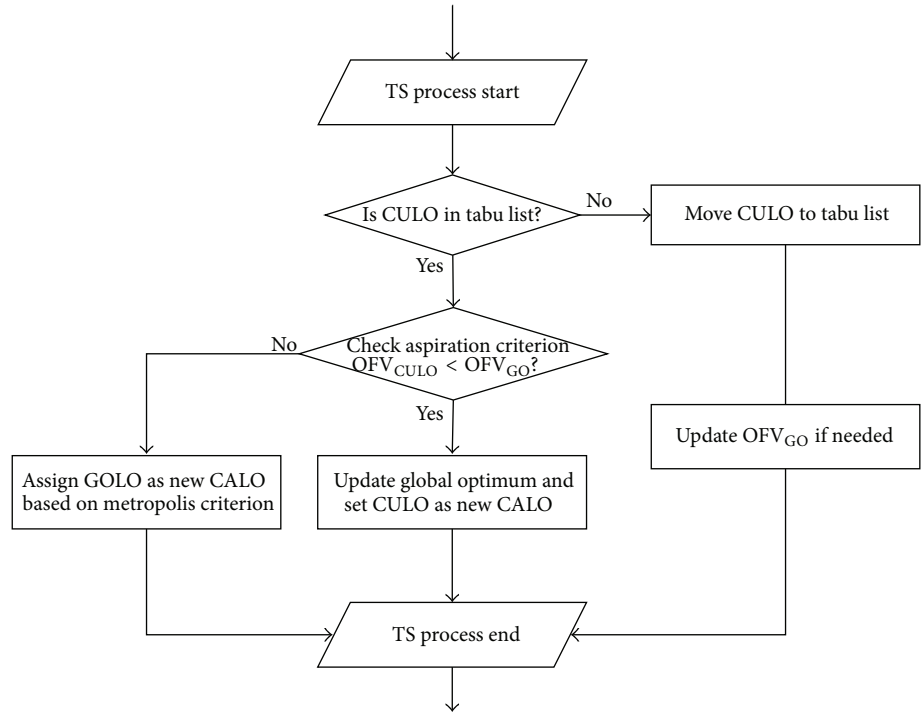


Figure 2: Flowchart of TS process in SATSO-GWT. The OFV_{GO} represents the current global optimal OFV, OFV_{CULO} represents the optimal OFV at current location, GOLO represents the global optimal location, CALO represents the candidate location, and CULO represents the current location.

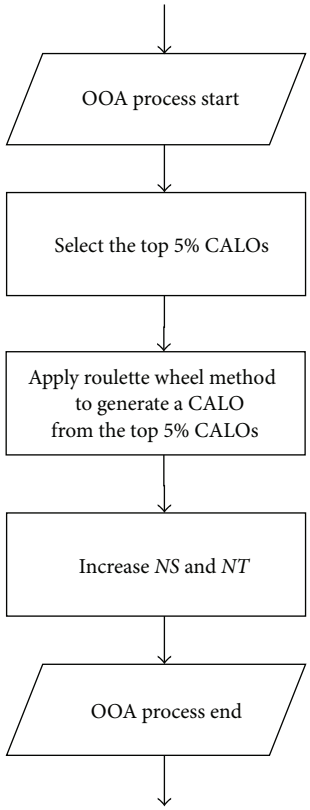


Figure 3: Flowchart of OOA in SATSO-GWT.

the monitoring points using (3) and the OFV corresponding to each set of trial solution is then computed using (4). Each CALO is regarded as one subdomain and the best combination of the source location and the release periods and concentrations in the monitoring points, i.e., the least objective function value at each CULO (hereinafter referred to as OFV_{CULO}), is recorded at each subdomain. The TS process (Figure 2) is then applied to determine whether the CULO is in tabu list. If not, the CULO is moved to tabu list and the OFV_{GO} is replaced by OFV_{CULO} if $OFV_{CULO} < OFV_{GO}$. Otherwise, check the aspiration criterion (i.e., $OFV_{CULO} < OFV_{GO}$). If $OFV_{CULO} < OFV_{GO}$, the OFV_{GO} is replaced by OFV_{CULO} and the CULO is set as new CALO. On the other hand, the global optimal location (GOLO) is assigned as new CALO based on Metropolis criterion defined as

$$P_L = \exp\left(\frac{OFV_{GO} - OFV_{CULO}}{T}\right), \tag{5}$$

where P_L is the acceptance probability of the trial location and T is the current temperature defined by SA. A random number ranging from zero to one is generated to compare with P_L. The GOLO will be rejected if P_L is less than the random number.

Totally, NT CALOs are generated at each temperature; therefore, NT sets of the combinations are obtained. After generating 3 times of CALOs, the top 5% best subdomains can be sifted by the OOA as demonstrated in Figure 3 The roulette wheel method is then applied so that

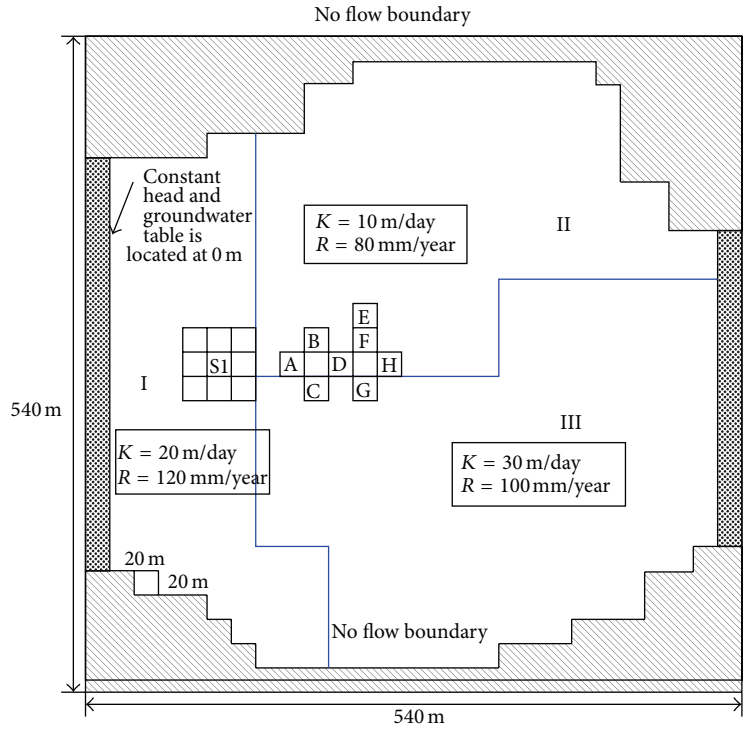


Figure 4: The groundwater flow system has an area of 540 m by 540 m and the domain is divided into three areas with different hydraulic conductivities and recharge rates. The real source is located at S1 and A to H represent the monitoring wells. The slash grids represent no flow boundary.

the better combinations (lower OFVs) regarding source release information have larger probability to be chosen. In reality, the real source location falls in the top 5% best combinations. The algorithm is terminated when the OFVs are less than 10^{-6} four times successively. Finally, the latest updated solution, including the estimated location and the release concentrations and time periods, is considered as the final solution.

3. Results and Discussion

3.1. Hypothetic Contamination Site and Identification Results. A hypothetic site shown in Figure 4 is designed to test the applicability of SATSO-GWT for a source identification problem. The domain of the site is divided into $27 \times 27 \times 4$ finite difference meshes in x-, y-, and z-directions. Both the grid width and length are 20 m and the grid height is 6 m. Thus, the total length and width of the site are both 540 m, and the aquifer thickness is 24 m. The site is heterogeneous and divided into three different areas with the hydraulic conductivities of 20 m/day, 10 m/day, and 30 m/day in areas I, II, and III, respectively. The aquifer porosity, specific storage, and hydraulic gradient are 0.3, $10^{-4}\,\mathrm{m}^{-1}$, and 0.009, respectively. The recharge rates are assumed to be 120 mm/year, 80 mm/year, and 100 mm/year in areas I, II, and III, respectively, in the first 180 days. The contaminant is assumed to be no decay and not adsorbed by the aquifer media. The dispersion coefficients in x-, y-, and z-directions are $40\,\mathrm{m}^2$/day, $10\,\mathrm{m}^2$/day, and $1\,\mathrm{m}^2$/day, respectively. The boundary conditions for the flow system are illustrated in Figure 4. The slash grids represent no flow boundary. The origin of the vertical coordinate is taken at the land surface. The source S1 is located at coordinates (110 m, 270 m, −9 m) and the rate of source release (Q) is $1\,\mathrm{m}^3$/day with the concentrations of 100 ppm and 50 ppm over first and second 180 days, respectively. There are seven unknowns to be determined including three coordinates of the source location and two release periods and release concentrations. Yeh et al. [15] mentioned that the number of sampling points should be greater than the number of unknowns. Accordingly, eight sampling points, i.e., points A to H shown in Figure 4, with various depths are considered. The A2 represents that the concentration measurement is sampled from second layer below the ground surface at point A. The concentration measurements at these sampling points are listed in Table 1. The groundwater transport model MF2K-GWT is utilized to generate the concentrations at these monitoring wells and the SATSO-GWT is used to determine the source information.

Before the source is identified, a block with $3 \times 3 \times 4$ meshes is delineated as a suspicious area which contains the contaminant source. Thus, there are 36 candidate sources within the block and one of the candidates is the real source. The lower and upper bounds of the release period are taken as 0 day and 400 days, respectively, and the lower and upper bounds of the release concentration are considered 0 ppm and 200 ppm, respectively. If the measures of release period and concentration have the accuracy to the first decimal place,

Table 1: The sampling points and concentration measurements.

Sampling point	Concentration measurement (ppm)
A2	$2.231E-01$
B1	$1.536E-01$
C2	$1.930E-01$
D4	$1.215E-01$
E3	$6.441E-02$
F2	$1.195E-01$
G1	$1.675E-01$
H3	$1.213E-01$

the feasible solution domain will be $36 \times 4000^2 \times 2000^2$. Such a solution space is very huge and poses a large computational burden to find the contaminant source information. Therefore, the OOA is adopted in SATSO-GWT in the identification process. Once the SATSO-GWT generates the CALOs for 3 times, the top 5% combinations with different source locations can be sifted. To state more specifically, there are 2 best locations ($36 \times 0.05 = 1.8 \approx 2$) that can be sifted. The obtained results of the sifted locations from eight different initial locations are listed in Table 2 indicating that the real source location (110 m, 270 m, −9 m) already falls within the top 2 best locations and, thus, the solution space is largely reduced. Note that the parameters *NS*, *NT*, initial temperature, and reduce temperature factor are taken as 20, 10, 0.5, and 0.7, respectively, in this case study.

Table 3 shows the identification results of the case study using SATS-GWT and SATSO-GWT. The same SA parameter values and initial location, i.e., at coordinates (290 m, 130 m, −21 m) are used for these two approaches. The SATS-GWT obtains correct source location but has deviated results for the release period and concentration. In contrast, the information of contaminant source is accurately identified by SATSO-GWT. Moreover, SATS-GWT takes about 1 day and 10 hours to obtain the result while SATSO-GWT only consumes about 12 hours and 36 minutes performing on a personal computer with Intel 3.3 G E3-1230v2 CPU and 16 GB RAM. From this table, the performance of SATSO-GWT is much superior to that of SATS-GWT.

To further assess the performance of SATSO-GWT, the following three cases are considered: (1) different initial guesses of source location; (2) various measurement errors in the monitoring points; and (3) a large suspicious area with various source release periods and concentrations.

3.2. Different Initial Guesses of Source Location. In this case, eight scenarios with different initial locations are considered. Eight different source locations, situated at the corners of the area, are chosen to investigate the influence of different initial locations. Table 4 shows the identified results of the source location as well as two release periods and concentrations. Figure 5 displays the temporal concentration distribution of eight scenarios observed at monitoring well A2. The predicted results exhibit excellent match with the observation

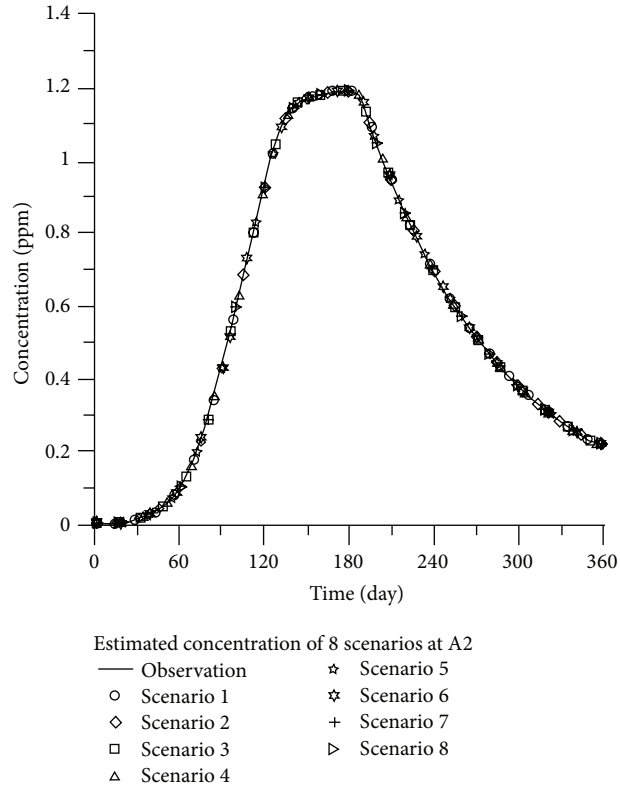


Figure 5: The temporal concentration distribution at monitor well A2 with 8 different initial guesses of source location.

data within 360 days. In these eight cases, the estimated source locations are all correct. In addition, the estimated release periods and concentrations are fairly good when compared with the real release data.

3.3. Measurement Errors in the Monitoring Points. The second case is to assess the performance of SATSO-GWT when the simulated sampling concentrations contain measurement errors. The disturbed observed concentrations are expressed as (Mahar and Datta [2])

$$C'_{i,\text{obs}} = C_{i,\text{obs}} \times \left(1 + \text{Er} \times \text{RD}_1\right), \tag{6}$$

where $C'_{i,\text{obs}}$ is the disturbed observed concentration, Er is defined as the level of measurement error, and RD_1 is a random standard normal deviate generated by the routine RNNOF of IMSL [66]. Three different values of Er, 1%, 5%, and 10%, are considered for this problem.

The predicted results shown in Table 5 indicate that the source locations of those three scenarios are all correctly identified. When Er = 1%, the optimal OFV is 0.490×10^{-7}. As Er = 10%, the OFV is 13.04×10^{-7}. The predicted release periods and concentrations are slightly deviated from the target values in scenario 3 but still are acceptable. Figure 6 shows the temporal concentration at monitoring well A2 predicted by SATSO-GWT. The results indicate that even though the sampling concentrations contain measurement errors whose level is up to 10%, the proposed SATSO-GWT

Table 2: Results of 8 scenarios for sifting the top two locations.

Scenario	Initial guess value	Sifted results			
	Guess source location (m)	First source location (m)	Current objective function value ($\times 10^{-5}$)	Second source location (m)	Current objective function value ($\times 10^{-5}$)
1	(250, 90, −3)	(110, 270, −9)	2.408	(90, 270, −9)	7.418
2	(250, 90, −21)	(110, 270, −9)	2.068	(90, 270, −9)	8.217
3	(250, 130, −3)	(90, 270, −9)	3.628	(110, 270, −9)	5.354
4	(250, 130, −21)	(110, 270, −9)	0.503	(90, 270, −9)	13.106
5	(290, 90, −3)	(90, 270, −9)	1.739	(110, 270, −9)	3.448
6	(290, 90, −21)	(110, 270, −9)	0.345	(90, 270, −9)	2.393
7	(290, 130, −3)	(110, 270, −9)	1.977	(90, 270, −9)	5.492
8	(290, 130, −21)	(90, 270, −9)	2.582	(110, 270, −9)	3.276

Note that the real source is located at (110 m, 270 m, −9 m).

Table 3: The identified results using SATS-GWT and SATSO-GWT.

Methodology	Source location (m)	Results					
		First release period (day)	First release concentration (ppm)	Second release period (day)	Second release concentration (ppm)	Computer time	Objective function value ($\times 10^{-9}$)
SATS-GWT	(110, 270, −9)	192.18	144.60	49.47	200.27	1 day 10 hours	1057.7
SATSO-GWT	(110, 270, −9)	180.19	99.90	179.58	50.02	12 hours 36 minutes	4.145

Table 4: Results of 8 scenarios with different initial guesses of source location.

Scenario	Initial guess value	Results					Objective function value ($\times 10^{-9}$)
	Guess source location (m)	Source location (m)	First release period (day)	First release concentration (ppm)	Second release period (day)	Second release concentration (ppm)	
1	(250, 90, −3)	(110, 270, −9)	178.90	100.14	180.04	49.99	6.035
2	(250, 90, −21)	(110, 270, −9)	180.25	99.65	179.41	49.92	6.053
3	(250, 130, −3)	(110, 270, −9)	177.38	100.91	180.14	50.01	7.483
4	(250, 130, −21)	(110, 270, −9)	179.86	99.99	180.11	49.97	2.031
5	(290, 90, −3)	(110, 270, −9)	180.01	99.92	180.10	50.07	2.165
6	(290, 90, −21)	(110, 270, −9)	180.14	99.80	179.55	49.91	4.520
7	(290, 130, −3)	(110, 270, −9)	179.38	99.76	178.53	49.78	9.155
8	(290, 130, −21)	(110, 270, −9)	180.19	99.90	179.58	50.02	4.145

Note that the real source is located at (110 m, 270 m, −9 m); real release concentration is 100 ppm over the first 180 days and 50 ppm over the second 180 days.

Table 5: Results of three scenarios when observed concentrations have measurement errors with different levels.

Scenario	Error level (%)	Results					Optimal objective function value ($\times 10^{-7}$)
		Source location (m)	First release period (day)	First release concentration (ppm)	Second release period (day)	Second release concentration (ppm)	
1	1	(110, 270, −9)	179.55	99.77	177.14	49.29	0.490
2	5	(110, 270, −9)	185.51	98.27	174.49	46.29	3.452
3	10	(110, 270, −9)	189.21	95.04	168.23	43.48	9.835

Note that the real source is located at (110 m, 270 m, −9 m); real release concentration is 100 ppm over the first 180 days and 50 ppm over the second 180 days.

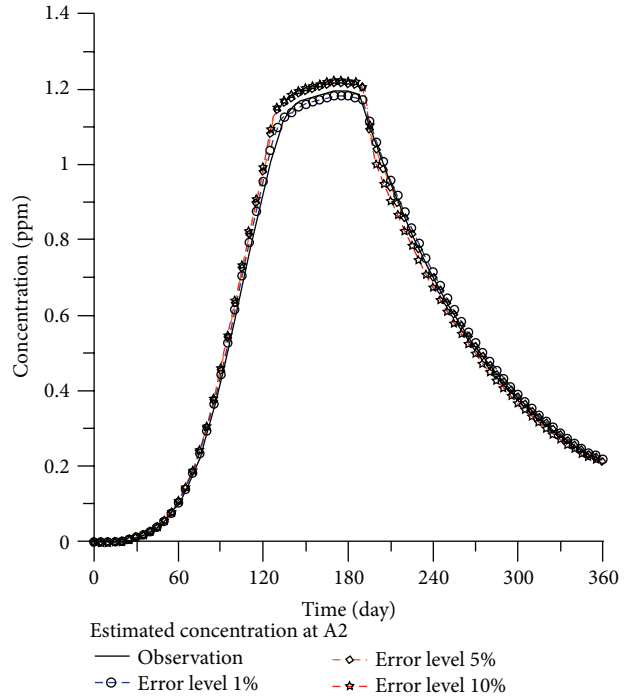


Figure 6: The temporal concentration distribution at monitor well A2 with 3 measurement error levels.

still gives fairly good results in reconstructing the release history and identifying the source location.

3.4. Larger Suspicious Area with Six Release Periods. In Section 3.1, it has been shown that SATSO-GWT can reduce the feasible solution domain based on OOA for a complex source information identification problem. Thus, the SATSO-GWT is applied to the case of a large suspicious area which has 100 candidate source locations (5 rows × 5 columns × 4 layers) delineated by the dashed lines as shown in Figure 7. This case considers six source release periods within a year and each period has an interval of two months. The concentrations in those six release periods are assumed to be 100 ppm, 200 ppm, 150 ppm, 50 ppm, 100 ppm, and 70 ppm. Therefore, there are fifteen unknowns involved in this case (i.e., three coordinates of the source location, six release periods, and six release concentrations). Thus, sixteen sampling data from wells A to H at 360 days and 390 days are considered and listed in Table 6.

The parameter *NT* associated with the generated locations by TS process at each temperature is taken as 25 to accommodate larger candidate locations. Because the number of the total CALOs is modified as 100, there are 5 best locations (5%) chosen by OOA. The upper part of Table 7 displays the top 5 best locations and the rank number 1 of sifted location exactly matches the real one. In Table 7, SATSO-GWT gives correct identification of source location and the predicted source release history is very close to the target one. This case has fifteen unknowns and the SATSO-GWT takes about 12 hours to obtain the results when performing on a personal computer with Intel 3.3 G E3-1230v2 CPU and 16 GB RAM. Obviously, the SATSO-GWT works very well in identifying the source information even when the suspicious area is large and the release pattern is rather irregular.

Table 6: The sampling points and concentration measurements in third case.

Sampling point	Concentration measurement (ppm)	
	At 360 (day)	At 390 (day)
A2	$3.467E-01$	$2.981E-01$
B1	$2.124E-01$	$1.997E-01$
C2	$2.882E-01$	$2.496E-01$
D4	$1.586E-01$	$1.553E-01$
E3	$9.521E-02$	$9.418E-02$
F2	$1.710E-01$	$1.608E-01$
G1	$2.103E-01$	$1.998E-01$
H3	$1.671E-01$	$1.587E-01$

4. Concluding Remarks

A novel identification approach, SATSO-GWT, is developed to combine the model MD2K-GWT for simulating the groundwater flow and pollutant transport MD2K-GWT with heuristic optimization approaches such as SA, TS, OOA, and roulette wheel method. This new approach is capable of simultaneously identifying the pollution source location and release history in a 3D, heterogeneous, and transient groundwater flow system. A hypothetic contamination site consisted of 27 × 27 × 4 finite difference meshes along with a suspicious area of having 3 × 3 × 4 meshes is designed to assess the capability of the present approach. The site is divided into three different areas and each area has different hydraulic conductivity and surface recharge rate. The contaminant source is continuously released over two periods with different concentrations. The present approach successfully identifies the source location and corresponding release periods and concentrations. Moreover, the results obtained from the same problem indicate that the performance of SATSO-GWT is much superior to that of SATS-GWT.

Three cases are designed to further assess the performance of SATSO-GWT. These are (1) different initial guesses of source location; (2) various measurement errors in the monitoring points; and (3) a large suspicious area with various source release periods and concentrations. The SATSO-GWT gives exact identification in source location and accurate predictions of release periods and concentrations in eight scenarios with different initial guess locations. In addition, the SATSO-GWT gives fairly good results when the sampling concentration having measurement errors, even when the error level is up to 10%. For a large suspicious area with six release periods and concentrations, the SATSO-GWT can also give excellent results which demonstrate its capability of

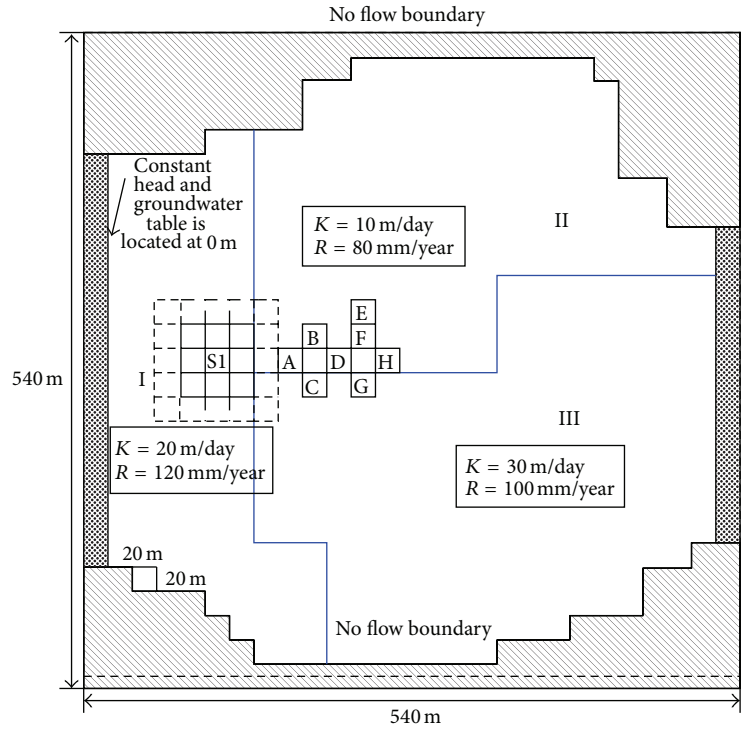


Figure 7: A larger suspicious areas delineated by the broken lines with totally 100 suspicious areas (5 rows × 5 columns × 4 layers). The hydrogeological conditions of the flow system are the same as those shown in Figure 4.

Table 7: Results of the larger suspicious areas with six release periods and concentrations.

(a)

Initial guess source location	Sifted results			Real source location (m)
	Rank	Sifted location (m)	Current objective function value ($\times 10^{-4}$)	
(150, 310, −21)	1	(110, 270, −9)	0.371	(110, 270, −9)
	2	(90, 270, −9)	3.277	
	3	(130, 270, −9)	4.650	
	4	(90, 270, −3)	10.61	
	5	(90, 250, −9)	12.77	

(b)

Final result								
Estimated source location	First release period (day)	Second release period (day)	Third release period (day)	Fourth release period (day)	Fifth release period (day)	Sixth release period (day)	Optimal objective function value ($\times 10^{-7}$)	Computer time
(110, 270, −9)	60.012	56.509	66.845	55.070	61.556	60.587		
	First release concentration (ppm)	Second release concentration (ppm)	Third release concentration (ppm)	Fourth release concentration (ppm)	Fifth release concentration (ppm)	Sixth release concentration (ppm)		
	105.53	194.59	147.29	56.852	97.929	70.829	8.191	12 hours

Note that the real release concentration is 100 ppm over the first 60 days, 200 ppm over the second 60 days, 150 ppm over the third 60 days, 50 ppm over the fourth 60 days, 100 ppm over the fifth 60 days, and 70 ppm over the sixth 60 days.

dealing with complex optimization problems. The SATSO-GWT has been shown to be an efficient tool in solving the complicated groundwater source identification problems.

Conflict of Interests

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

Acknowledgments

This study was partly supported by the Ministry of Science and Technology under the Grants NSC 101-2221-E-009-105-MY2 and 102-2221-E-009-072-MY2. The authors would like to thank the editor and two anonymous reviewers for their valuable and constructive comments.

References

[1] J. Atmadja and A. C. Bagtzoglou, "State of the art report on mathematical methods for groundwater pollution source identification," *Environmental Forensics*, vol. 2, no. 3, pp. 205–214, 2001.

[2] P. S. Mahar and B. Datta, "Optimal identification of groundwater pollution sources and parameter estimation," *Journal of Water Resources Planning and Management*, vol. 127, no. 1, pp. 20–29, 2001.

[3] B. Datta, "Discussion of "Identification of contaminant source location and release history in aquifers" by Mustafa M. Aral, Jiabao Guan, and Morris L. Maslia," *Journal of Hydrologic Engineering*, vol. 7, no. 5, pp. 399–400, 2002.

[4] R. M. Singh and B. Datta, "Artificial neural network modeling for identification of unknown pollution sources in groundwater with partially missing concentration observation data," *Water Resources Management*, vol. 21, no. 3, pp. 557–572, 2007.

[5] Z. Li and X.-Z. Mao, "Global multiquadric collocation method for groundwater contaminant source identification," *Environmental Modelling and Software*, vol. 26, no. 12, pp. 1611–1621, 2011.

[6] M. Jha and B. Datta, "Three-dimensional groundwater contamination source identification using adaptive simulated annealing," *Journal of Hydrologic Engineering*, vol. 18, no. 3, pp. 307–317, 2013.

[7] A. Y. Sun, "A robust geostatistical approach to contaminant source identification," *Water Resources Research*, vol. 43, no. 2, Article ID W02418, 2007.

[8] S. M. Gorelick, B. Evans, and I. Remson, "Identifying sources of groundwater pollution: an optimization approach," *Water Resources Research*, vol. 19, no. 3, pp. 779–790, 1983.

[9] J. C. Hwang and R. M. Koerner, "Groundwater pollution source identification from limited monitoring well data. Part 1. Theory and feasibility," *Journal of Hazardous Materials*, vol. 8, no. 2, pp. 105–119, 1983.

[10] National Research Council, *Groundwater Models—Scientific and Segulatory Applications*, National Academy Press, Washington, DC, USA, 1990.

[11] A. C. Bagtzoglou, D. E. Dougherty, and A. F. B. Tompson, "Application of particle methods to reliable identification of groundwater pollution sources," *Water Resources Management*, vol. 6, no. 1, pp. 15–23, 1992.

[12] A. Sciortino, T. C. Harmon, and W. W.-G. Yeh, "Inverse modeling for locating dense nonaqueous pools in groundwater under steady flow conditions," *Water Resources Research*, vol. 36, no. 7, pp. 1723–1735, 2000.

[13] G. Mahinthakumar and M. Sayeed, "Hybrid genetic algorithm—local search methods for solving groundwater source identification inverse problems," *Journal of Water Resources Planning and Management*, vol. 131, no. 1, pp. 45–57, 2005.

[14] E. Milnes and P. Perrochet, "Simultaneous identification of a single pollution point-source location and contamination time under known flow field conditions," *Advances in Water Resources*, vol. 30, no. 12, pp. 2439–2446, 2007.

[15] H. D. Yeh, T. H. Chang, and Y. C. Lin, "Groundwater contaminant source identification by a hybrid heuristic approach," *Water Resources Research*, vol. 43, no. 9, 2007.

[16] B. Datta, D. Chakrabarty, and A. Dhar, "Simultaneous identification of unknown groundwater pollution sources and estimation of aquifer parameters," *Journal of Hydrology*, vol. 376, no. 1-2, pp. 48–57, 2009.

[17] M. T. Ayvaz, "A linked simulation-optimization model for solving the unknown groundwater pollution source identification problems," *Journal of Contaminant Hydrology*, vol. 117, no. 1–4, pp. 46–59, 2010.

[18] B. J. Wagner, "Simultaneous parameter estimation and contaminant source characterization for coupled groundwater flow and contaminant transport modelling," *Journal of Hydrology*, vol. 135, no. 1–4, pp. 275–303, 1992.

[19] T. H. Skaggs and Z. J. Kabala, "Recovering the release history of a groundwater contaminant," *Water Resources Research*, vol. 30, no. 1, pp. 71–79, 1994.

[20] T. H. Skaggs and Z. J. Kabala, "Recovering the history of a groundwater contaminant plume: method of quasi-reversibility," *Water Resources Research*, vol. 31, no. 11, pp. 2669–2673, 1995.

[21] T. H. Skaggs and Z. J. Kabala, "Limitations in recovering the history of a groundwater contaminant plume," *Journal of Contaminant Hydrology*, vol. 33, no. 3-4, pp. 347–359, 1998.

[22] A. D. Woodbury and T. J. Ulrych, "Minimum relative entropy inversion: theory and application to recovering the release history of a groundwater contaminant," *Water Resources Research*, vol. 32, no. 9, pp. 2671–2681, 1996.

[23] M. F. Snodgrass and P. K. Kitanidis, "A geostatistical approach to contaminant source identification," *Water Resources Research*, vol. 33, no. 4, pp. 537–546, 1997.

[24] A. Woodbury, E. Sudicky, T. J. Ulrych, and R. Ludwig, "Three-dimensional plume source reconstruction using minimum relative entropy inversion," *Journal of Contaminant Hydrology*, vol. 32, no. 1-2, pp. 131–158, 1998.

[25] L. Chongxuan and W. P. Ball, "Application of inverse methods to contaminant source identification from aquitard diffusion profiles at Dover AFB, Delaware," *Water Resources Research*, vol. 35, no. 7, pp. 1975–1985, 1999.

[26] R. M. Neupauer and J. L. Wilson, "Adjoint method for obtaining backward-in-time location and travel time probabilities of a conservative groundwater contaminant," *Water Resources Research*, vol. 35, no. 11, pp. 3389–3398, 1999.

[27] R. M. Neupauer and J. L. Wilson, "Adjoint-derived location and travel time probabilities for a multidimensional groundwater system," *Water Resources Research*, vol. 37, no. 6, pp. 1657–1668, 2001.

[28] R. M. Neupauer, B. Borchers, and J. L. Wilson, "Comparison of inverse methods for reconstructing the release history of a groundwater contamination source," *Water Resources Research*, vol. 36, no. 9, pp. 2469–2475, 2000.

[29] J. Atmadja and A. C. Bagtzoglou, "Pollution source identification in heterogeneous porous media," *Water Resources Research*, vol. 37, no. 8, pp. 2113–2125, 2001.

[30] A. C. Bagtzoglou and J. Atmadja, "The Marching-Jury backward beam equation and quasi-reversibility methods for hydrologic inversion: application to contaminant plume spatial distribution recovery," *Water Resources Research*, vol. 39, no. 2, pp. 1038–1051, 2003.

[31] I. Butera and M. G. Tanda, "A geostatistical approach to recover the release history of groundwater pollutants," *Water Resources Research*, vol. 39, article 1372, no. 12, 2003.

[32] S. Shlomi and A. M. Michalak, "A geostatistical framework for incorporating transport information in estimating the distribution of a groundwater contaminant plume," *Water Resources Research*, vol. 43, no. 3, Article ID W03412, 2007.

[33] N. M. Muhammad, K.-Y. Kim, C.-H. Huang, and S. Kim, "Groundwater contaminant boundary input flux estimation in a two-dimensional aquifer," *Journal of Industrial and Engineering Chemistry*, vol. 16, no. 1, pp. 106–114, 2010.

[34] A. D. Koussis, K. Mazi, S. Lykoudis, and A. A. Argiriou, "Reverse flood routing with the inverted Muskingum storage routing scheme," *Natural Hazards and Earth System Science*, vol. 12, no. 1, pp. 217–227, 2012.

[35] M. M. Aral and J. Guan, "Genetic algorithms in search of groundwater pollution sources," *Advances in Groundwater Pollution Control and Remediation*, vol. 9, pp. 347–369, 1996.

[36] A. C. Bagtzoglou, "On the nonlocality of reversible-time particle tracking methods," *Environmental Forensics*, vol. 4, no. 3, pp. 215–225, 2003.

[37] A. C. Bagtzoglou and J. Atmadja, "Mathematical methods for hydrologic inversion: the case of pollution source identification," in *Water Pollution*, vol. 3 of *The Handbook of Environmental Chemistry*, pp. 65–96, Springer, Berlin, Germany, 2005.

[38] R. M. Neupauer and R. Lin, "Identifying sources of a conservative groundwater contaminant using backward probabilities conditioned on measured concentrations," *Water Resources Research*, vol. 42, no. 3, Article ID W03424, 2006.

[39] A. Y. Sun, S. L. Painter, and G. W. Wittmeyer, "A constrained robust least squares approach for contaminant release history identification," *Water Resources Research*, vol. 42, no. 4, Article ID W04414, 2006.

[40] A. Y. Sun, S. L. Painter, and G. W. Wittmeyer, "A robust approach for iterative contaminant source location and release history recovery," *Journal of Contaminant Hydrology*, vol. 88, no. 3-4, pp. 181–196, 2006.

[41] R. Ababou, A. C. Bagtzoglou, and A. Mallet, "Anti-diffusion and source identification with the "RAW" scheme: a particle-based censored random walk," *Environmental Fluid Mechanics*, vol. 10, no. 1, pp. 41–76, 2010.

[42] I. Butera, M. G. Tanda, and A. Zanini, "Simultaneous identification of the pollutant release history and the source location in groundwater by means of a geostatistical approach," *Stochastic Environmental Research and Risk Assessment*, vol. 27, no. 5, pp. 1269–1280, 2013.

[43] M. M. Aral, J. Guan, and M. L. Maslia, "Identification of contaminant source location and release history in aquifers," *Journal of Hydrologic Engineering*, vol. 6, no. 3, pp. 225–234, 2001.

[44] R. M. Neupauer and J. L. Wilson, "Backward probability model using multiple observations of contamination to identify groundwater contamination sources at the Massachusetts Military Reservation," *Water Resources Research*, vol. 41, no. 2, pp. 1–14, 2005.

[45] Y. C. Ho, R. S. Sreenivas, and P. Vakili, "Ordinal optimization of DEDS," *Discrete Event Dynamic Systems*, vol. 2, no. 1, pp. 61–88, 1992.

[46] Y. Ho and M. E. Larson, "Ordinal optimization approach to rare event probability problems," *Discrete Event Dynamic Systems: Theory and Applications*, vol. 5, no. 2-3, pp. 281–301, 1995.

[47] T. W. E. Lau and Y.-C. Ho, "Universal alignment probabilities and subset selection for ordinal optimization," *Journal of Optimization Theory and Applications*, vol. 93, no. 3, pp. 455–489, 1997.

[48] Y.-C. Ho, "An explanation of ordinal optimization: soft computing for hard problems," *Information Sciences*, vol. 113, no. 3-4, pp. 169–192, 1999.

[49] M. C. Fu, "Optimization for simulation: theory vs. practice," *INFORMS Journal on Computing*, vol. 14, no. 3, pp. 192–215, 2002.

[50] L. F. Konikow, D. J. Goode, and G. Z. Hornberger, "A three-dimensional method of characteristics solute-transport model (MOC3D)," U.S. Geological Survey Water-Resources Investigations Report 96-4267, 1996.

[51] A. W. Harbaugh, E. R. Banta, M. C. Hill, and M. G. McDonald, "MODFLOW-2000, the U.S. Geological survey modular ground-water model—user guide to modularization concepts and the ground-water flow process," U.S. Geological Survey, Open File Rep, 2000.

[52] W. H. Press, B. P. Flannery, S. A. Teukolsky, and W. T. Vetterling, *Numerical Recipes*, Cambridge University Press, Cambridge, UK, 2nd edition, 1992.

[53] N. Metropolis, A. W. Rosenbluth, M. N. Rosenbluth, A. H. Teller, and E. Teller, "Equation of state calculations by fast computing machines," *The Journal of Chemical Physics*, vol. 21, no. 6, pp. 1087–1092, 1953.

[54] Y.-C. Huang and H.-D. Yeh, "The use of sensitivity analysis in on-line aquifer parameter estimation," *Journal of Hydrology*, vol. 335, no. 3-4, pp. 406–418, 2007.

[55] H.-D. Yeh and Y.-J. Chen, "Determination of skin and aquifer parameters for a slug test with wellbore-skin effect," *Journal of Hydrology*, vol. 342, no. 3-4, pp. 283–294, 2007.

[56] H. Yeh, Y. Lin, and Y. Huang, "Parameter identification for leaky aquifers using global optimization methods," *Hydrological Processes*, vol. 21, no. 7, pp. 862–872, 2007.

[57] H. Yeh and Y. Lin, "Pipe network system analysis using simulated annealing," *Journal of Water Supply: Research and Technology—AQUA*, vol. 57, no. 5, pp. 317–327, 2008.

[58] Y.-C. Lin and H.-D. Yeh, "Identifying groundwater pumping source information using optimization approach," *Hydrological Processes*, vol. 22, no. 16, pp. 3010–3019, 2008.

[59] F. Glover, "Future paths for integer programming and links to artificial intelligence," *Computers & Operations Research*, vol. 13, no. 5, pp. 533–549, 1986.

[60] C. Zheng and P. Wang, "Parameter structure identification using tabu search and simulated annealing," *Advances in Water Resources*, vol. 19, no. 4, pp. 215–224, 1996.

[61] C. Tung and C. Chou, "Pattern classification using tabu search to identify the spatial distribution of groundwater pumping," *Hydrogeology Journal*, vol. 12, no. 5, pp. 488–496, 2004.

[62] X. Guan, Y. C. Ho, and F. Lai, "An ordinal optimization based bidding strategy for electric power suppliers in the daily energy market," *IEEE Transactions on Power Systems*, vol. 16, no. 4, pp. 788–797, 2001.

[63] S. Lin, Y. Ho, and C. Lin, "An ordinal optimization theory-based algorithm for solving the optimal power flow problem with discrete control variables," *IEEE Transactions on Power Systems*, vol. 19, no. 1, pp. 276–286, 2004.

[64] Y. Liu, J. Chen, and M. Xie, "Distribution network planning based on the ordinal optimization theory," *Automation of Electric Power Systems*, vol. 30, no. 22, pp. 21–24, 2006.

[65] S.-Y. Lin and S.-C. Horng, "Application of an ordinal optimization algorithm to the wafer testing process," *IEEE Transactions on Systems, Man, and Cybernetics A: Systems and Humans*, vol. 36, no. 6, pp. 1229–1234, 2006.

[66] IMSL, *Fortran Library User's Guide Gtat/Library*, vol. 2, Visual Numerics, Houston, Tex, USA, 2003.

Hindawi Publishing Corporation
Mathematical Problems in Engineering
Volume 2014, Article ID 464056, 8 pages
http://dx.doi.org/10.1155/2014/464056

Research Article

Trajectory Evaluation of Rotor-Flying Robots Using Accurate Inverse Computation Based on Algorithm Differentiation

Yuqing He, Yingjun Zhou, and Jianda Han

State Key Laboratory of Robotics, Shenyang Institute of Automation, Chinese Academy of Sciences, Shenyang 110016, China

Correspondence should be addressed to Yuqing He; heyuqing@sia.cn

Received 26 March 2014; Revised 9 June 2014; Accepted 11 June 2014; Published 13 August 2014

Academic Editor: Fatih Yaman

Autonomous maneuvering flight control of rotor-flying robots (RFR) is a challenging problem due to the highly complicated structure of its model and significant uncertainties regarding many aspects of the field. As a consequence, it is difficult in many cases to decide whether or not a flight maneuver trajectory is feasible. It is necessary to conduct an analysis of the flight maneuvering ability of an RFR prior to test flight. Our aim in this paper is to use a numerical method called algorithm differentiation (AD) to solve this problem. The basic idea is to compute the internal state (i.e., attitude angles and angular rates) and input profiles based on predetermined maneuvering trajectory information denoted by the outputs (i.e., positions and yaw angle) and their higher-order derivatives. For this purpose, we first present a model of the RFR system and show that it is flat. We then cast the procedure for obtaining the required state/input based on the desired outputs as a static optimization problem, which is solved using AD and a derivative based optimization algorithm. Finally, we test our proposed method using a flight maneuver trajectory to verify its performance.

1. Introduction

The autonomous rotor-flying robot (RFR) is one of the frontier research topics in the field of robotics. Extensive research has been conducted on issues related to RFR, including flight control [1, 2], path/trajectory planning [3, 4], and intelligent navigation [5, 6].

Flight maneuvering is a very important ability for an RFR system because it can greatly extend the system's field of applications. However, the flight maneuver control of an RFR is a challenging and absorbing problem in the field of autonomous RFR research [2]. This can be explained from the following two points. On the one hand, the model of an RFR system is an extremely complicated structure [7, 8], designing a suitable controller for which is very difficult. On the other hand, flight maneuvering requires that both the interval states (i.e., the attitude angles and the angular rates) and the control surfaces to have a large scope variation (or even approach the largest permitted value) [9]. This may produce a great deal of uncertainty and thus presents that this flight maneuvering is dangerous and difficult to be implemented and thus requires a much more robust controller than usual [10].

Most work on flight maneuver control, such as [11, 12], is based on successful flight demonstrations and grounded in learning techniques used to update the control laws. These methods have successfully pushed the envelope of what is possible with autonomous control, but they lack performance guarantees and are limited to situations where safety is not a critical concern. Thus, it is very important to assess the flight maneuvering ability of an RFR system prior to designing a controller and conducting tests. Some research has been conducted in this area. For instance, research in [2] is based on reachability analysis and explores aerobatic maneuvers and multiple vehicle collision avoidance of the RFR systems. This involves designing a sequence of modes and conducting safe transition from one to the next. However, reachability analysis is very time consuming, due to which the proposals in [2] are viable only for simplified models. Similarly, [9] involves a feasibility analysis of planned trajectories by using a greatly simplified dynamics model (without considering angular dynamics). Both the positions and attitude angles are predetermined and then used to evaluate the feasibility of the planned trajectories. Although the algorithm is simple and easily implemented, it is unfit for many planning algorithms

since for which it is difficult to obtain the positions and attitude angles simultaneously.

In this paper, based on a full state dynamics model, we investigate the feasibility analysis problem of a flight maneuvering trajectory. This is actually an inverse computation problem, that is, to compute both the interval states and the inputs based on predetermined position, yaw angle, and their higher-order derivatives. In our work, we model it as a static optimization problem and then use a derivative-based optimization algorithm to obtain an accurate solution of it. We use algorithm differentiation (AD) to obtain precise higher-order derivatives of the nonlinear cost function. The advantages of our proposed algorithm are as follows: (1) only with desired positions and their derivatives, which is the case for many planning algorithm, the feasibility of the trajectories (including inner states and inputs) can be evaluated; (2) inverse computation can be very accurate because AD is a numerical algorithm that can precisely obtain the derivative of a nonlinear function. Thus, an accurate pointwise numerical feasibility analysis of planned trajectories becomes possible.

The remainder of this paper is organized as follows. We first present a model of the RFR system and show that it is flat by taking the three positions and the yaw angle as a group of flat outputs. This allows us to model the trajectory evaluation problem as an optimization problem by taking into account the dynamical model, which is solved by using the AD algorithm along with a derivative-based optimization algorithm. Finally, in order to test our proposed method, we use it to analyze a flight maneuvering trajectory.

2. Dynamical Model of the RFR System

Usually, the complete dynamical model of an RFR system can be divided into three parts: the actuator dynamics, the aerodynamics, and the rigid body dynamics. In this paper, only the rigid body dynamics as shown in the following equation (1) is considered [3]:

$$\begin{bmatrix} \dot{p} \\ \dot{v}^p \\ \dot{\Theta} \\ \dot{\omega}^b \end{bmatrix} = \begin{bmatrix} v^p \\ \frac{1}{m} R f^b \\ \Psi \omega^b \\ J^{-1}\left(\tau^b - \omega^b \times J\omega^b\right) \end{bmatrix}, \tag{1}$$

where $p = [x\ y\ z]^T \in R^3$ and $v^p = [\dot{x}\ \dot{y}\ \dot{z}] \in R^3$ are translational position and velocity vector of the RFR in the inertia frame; R is the rotation matrix between the body frame and the inertia frame; ω^b is angular velocity vector; $\Theta = [\phi\ \theta\ \psi]^T$ is Euler angle vector; m and J are, respectively, the mass and inertia matrix of the RFR; Ψ is the transformation matrix from angular velocity vector in body frame to angular velocity vector in inertia frame; and f^b and τ^b are force and moment of the RFR presented in body frame. Different kinds of RFR have very different aerodynamics, that is, the relationship between the force/moment and the control surfaces. Here we focus on the quadracopter flying robot, whose driving forces and moments can be denoted as follows:

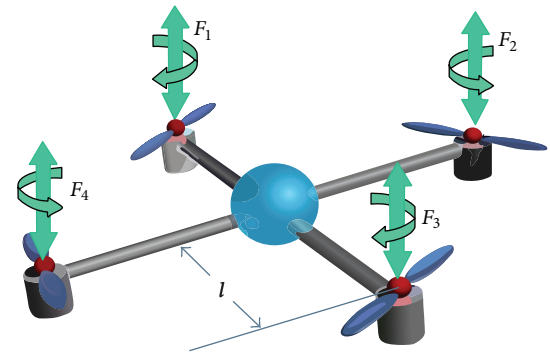


Figure 1: Sketch of quadrocopter.

$$f^b = \begin{bmatrix} f_x \\ f_y \\ f_z \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ T_M \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ F_1 + F_2 + F_3 + F_4 \end{bmatrix},$$
$$\tau^b = \begin{bmatrix} M_x \\ M_y \\ M_z \end{bmatrix} = \begin{bmatrix} l(F_4 - F_2) \\ l(F_3 - F_1) \\ l(F_1 - F_2 + F_3 - F_4) \end{bmatrix}, \tag{2}$$

where F_1, F_2, F_3, and F_4 are force produced by four rotors, respectively, and l is the distance between each rotor center and the RFR's center of gravity (see Figure 1).

Thus, define inputs as

$$u = \begin{bmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \end{bmatrix} \triangleq \begin{bmatrix} \ddot{T}_M \\ M_x \\ M_y \\ M_z \end{bmatrix}. \tag{3}$$

System (1) can be rewritten as

$$\begin{aligned} \ddot{x} &= \frac{\cos\psi \sin\theta \cos\varphi + \sin\psi \sin\varphi}{m} T_M, \\ \ddot{y} &= \frac{\sin\psi \sin\theta \cos\varphi - \sin\varphi \cos\psi}{m} T_M, \\ \ddot{z} &= -g + \frac{\cos\theta \cos\varphi}{m} T_M, \\ \ddot{\varphi} &= \frac{I_y - I_z}{I_x} \dot{\theta}\dot{\psi} + \frac{1}{I_x} u_2, \\ \ddot{\theta} &= \frac{I_z - I_x}{I_y} \dot{\varphi}\dot{\psi} + \frac{1}{I_y} u_3, \\ \ddot{\psi} &= \frac{I_x - I_y}{I_z} \dot{\theta}\dot{\varphi} + \frac{1}{I_z} u_4, \\ \ddot{T}_M &= u_1. \end{aligned} \tag{4}$$

It can be easily shown that system (4) is input-to-state feedback linearizable through defining the following state vector:

$$\left[x \;\; y \;\; z \;\; \dot{x} \;\; \dot{y} \;\; \dot{z} \;\; \ddot{x} \;\; \ddot{y} \;\; \ddot{z} \;\; \dddot{x} \;\; \dddot{y} \;\; \dddot{z} \;\; \psi \;\; \dot{\psi}\right]^T. \tag{5}$$

From (1) and (2), we have

$$\begin{aligned}
\mathbf{p}^{(4)} &= -\frac{1}{m}R\left[\boldsymbol{\omega}^b \times \left(\boldsymbol{\omega}^b \times \begin{bmatrix} 0 \\ 0 \\ T_M \end{bmatrix}\right)\right] + \frac{1}{m}R\begin{bmatrix} 0 & -T_M & 0 \\ T_M & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \\
&\quad \times J^{-1}\left(\tau^b - \boldsymbol{\omega}^b \times J\boldsymbol{\omega}^b\right) \\
&\quad - \frac{2}{m}R\left\{\boldsymbol{\omega}^b \times \begin{bmatrix} 0 & 0 & \dot{T}_M \end{bmatrix}^T\right\} - \frac{1}{m}R[0 \;\; 0 \;\; 1]^T\ddot{T}_M, \\
\ddot{\psi} &= -\begin{bmatrix} 0 & \dfrac{\sin\phi}{\cos\theta} & \dfrac{\cos\phi}{\cos\theta} \end{bmatrix} J^{-1}\left(\boldsymbol{\omega}^b \times J\boldsymbol{\omega}^b - \tau^b\right) \\
&\quad + \frac{\left(q\dot{\phi}\cos\phi - r\dot{\phi}\sin\phi\right)}{\cos\theta} \\
&\quad \times \left[p + 2q\sin\phi\tan\theta + 2r\cos\phi\tan\theta\right]
\end{aligned} \tag{6}$$

and the system can be transformed into the following linear form:

$$\begin{aligned}
p^{(4)} &= \begin{bmatrix} x^{(4)} \\ y^{(4)} \\ z^{(4)} \end{bmatrix} = \begin{bmatrix} v_1 \\ v_2 \\ v_3 \end{bmatrix}, \\
\ddot{\psi} &= v_4
\end{aligned} \tag{7}$$

with the feedback linearization controller

$$u = \begin{bmatrix} g_2 \\ g_3 \end{bmatrix}^{-1}\left(\begin{bmatrix} v_1 \\ v_2 \\ v_3 \\ v_4 \end{bmatrix} - \begin{bmatrix} f_2 \\ f_3 \end{bmatrix}\right), \tag{8}$$

where

$$\begin{aligned}
f_2 &= -\frac{2}{m}\dot{T}_M R\left(\omega^b \times \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}\right) \\
&\quad - \frac{1}{m}T_M R\left[\omega^b \times \left(\omega^b \times \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}\right)\right] \\
&\quad - \frac{1}{m}T_M R\left\{\left[J^{-1}\left(\omega^b \times J\omega^b\right)\right] \times \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}\right\}; \\
g_2 &= \left[-\frac{1}{m}R[0 \;\; 0 \;\; 1]^T \;\; -\frac{T_M}{m}R\begin{bmatrix} 0 & 1 & 0 \\ -1 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} J^{-1}\right]; \\
f_3 &= -\begin{bmatrix} 0 & \dfrac{\sin\phi}{\cos\theta} & \dfrac{\cos\phi}{\cos\theta} \end{bmatrix} J^{-1}\left(\omega^b \times J\omega^b\right) \\
&\quad + \left(q\cos\phi - r\sin\phi\right) \\
&\quad \times \sec\theta\left[p + 2q\sin\phi\tan\theta + 2r\cos\phi\tan\theta\right]; \\
g_3 &= \left[0 \;\; \begin{bmatrix} 0 & \dfrac{\sin\phi}{\cos\theta} & \dfrac{\cos\phi}{\cos\theta} \end{bmatrix} J^{-1}\right].
\end{aligned} \tag{9}$$

That means system (1) is flat and that the $[p^T, \psi]$ are the corresponding flat outputs.

3. Automatic Differential Algorithm Based Inverse Computation

AD is a numerical method that is effective to accurately compute the derivatives of some complicated nonlinear function. In this section, we will briefly introduce how to conduct the inverse computation of a nonlinear flat system using AD algorithm [13].

Consider the following nonlinear system:

$$\begin{aligned}
\dot{x} &= f(x), \\
y &= h(x).
\end{aligned} \tag{10}$$

It has been shown that the inverse computation problem of system (10), that is, to obtain a state vector x_d with $h(x_d) = y_d$ (a predefined output), is solvable if $y = h(x)$ is a flat output of system (1) [14].

Derivatives of y with respect to x are necessary to effectively solve the preceding inverse problem. In the following contents of this section, the basic idea on how to obtain $\partial y/\partial x$ using AD algorithm will be introduced.

For this purpose, we firstly use the following Taylor series to denote the state vector $x(t)$:

$$x(t) = x_0 + x_1 t + x_2 t^2 + \cdots + x_d t^d. \tag{11}$$

Provided that $f(*)$ is d times continuously differentiable, $z = f(x)$ can also be expressed by Taylor series as follows:

$$z(t) = z_0 + z_1 t + z_2 t^2 + \cdots + z_d t^d + \mathrm{O}\left(t^{d+1}\right). \tag{12}$$

It has been shown that each Taylor coefficient z_i ($i = 0, 1, \ldots, d$) is uniquely determined by the coefficients x_i ($i = 0, 1, \ldots, d$) as follows [15, 16]:

$$\begin{aligned}
z_0 &= f(x_0) \\
z_1 &= \frac{\partial f(x_0)}{\partial x_0}x_1 \\
z_2 &= \frac{\partial f(x_0)}{\partial x_0}x_2 + \frac{1}{2}\frac{\partial^2 f(x_0)}{\partial x_0 \partial x_0}x_1 x_1 \\
&\vdots
\end{aligned} \tag{13}$$

Thus, if x_i $(i = 0, 1, \ldots, d)$ in (11) are preknown, all z_i $(i = 0, 1, \ldots, d)$ can be obtained based on some derivatives, which can be computed by basic "forward mode" of AD [13]. Similarly, the following A_{j-i} can also be derived using "reverse mode" of AD [13–15]:

$$A_{j-i}\left(j \geq i\right) \triangleq \frac{\partial z_{j-i}}{\partial x_0}. \tag{14}$$

Furthermore, based on (10), we have

$$\begin{aligned} z\left(t\right) &= z_0 + z_1 t + z_2 t^2 + \cdots + z_d t^d + \mathrm{O}\left(t^{d+1}\right) \\ &= f\left(x\right) = \dot{x} = x_1 + 2x_2 t + \cdots + d x_d t^{d-1}; \end{aligned} \tag{15}$$

that is,

$$z_i \equiv \left(i+1\right) x_{i+1}, \quad i = 0, \ldots, d-1. \tag{16}$$

Through (16) and (13), we can easily compute all z_i based on x_0.

Define a new matrix as

$$B_k \triangleq \frac{dx_{k+1}}{dx_0} = \frac{1}{k+1}\frac{dz_k}{dx_0} = \frac{1}{k+1}\sum_{j=0}^{k}\frac{\partial z_k}{\partial x_j}\frac{dx_j}{dx_0}. \tag{17}$$

From [15], we have

$$\frac{\partial z_k}{\partial x_j} = \frac{\partial z_{k-j}}{\partial x_0}. \tag{18}$$

Thus, B_k can be iteratively obtained as follows:

$$B_k = \frac{1}{k+1}\sum_{j=0}^{k}\frac{\partial z_{k-j}}{\partial x_0}\frac{dx_j}{dx_0} = \frac{1}{k+1}\left(A_k + \sum_{j=0}^{k} A_{k-j}B_{j-1}\right). \tag{19}$$

Furthermore, if we denote $y(t)$ as

$$y\left(t\right) = y_0 + y_1 t + y_2 t^2 + \cdots + y_d t^d + \mathrm{O}\left(t^{d+1}\right), \tag{20}$$

then the same process can be used to compute y through the second equation of (10); that is, with a preknown x_0, compute y_i by forward mode and compute the following derivatives C_{j-i} by reverse mode:

$$C_{j-i} \triangleq \frac{\partial y_j}{\partial x_i}. \tag{21}$$

Also, it is not difficult to show that the derivatives of output with respect to the states can be denoted as follows:

$$D_k \triangleq \frac{dy_k}{dx_0} = \sum_{j=0}^{k}\frac{\partial y_k}{\partial x_j}\frac{dx_j}{dx_0} = C_k + \sum_{j=1}^{k} C_{k-j}B_{j-1}. \tag{22}$$

Up to now, we have shown that, with a predefined state x_0, both the corresponding outputs $y(x_0)$ and its higher-order derivatives with respect to time, as well as the derivatives with respect to state x_0 (22), can be obtained through AD algorithm. These results can be further used to conduct the inverse computation of system (1), and the detailed steps are given as follows.

AD Based Inverse Computation Algorithm

Initialization. d (the highest order of the outputs' derivatives); γ (step length of searching algorithm); $k = 0$ (iterative number).

Step 1. Compute the output vector $Y(x_k)$ using forward mode of AD algorithm at each time instant as a nonlinear map; that is,

$$\Theta : x^k \longrightarrow Y\left(x^k\right) \triangleq \begin{pmatrix} y_0\left(x^k\right) \\ \vdots \\ y_d\left(x^k\right) \end{pmatrix}. \tag{23}$$

Step 2. Based on (22), AD can be used to obtain the partial derivative of Y with respect to x_0; that is,

$$\Theta'\left(x^k\right) = \frac{\partial Y}{\partial x^k} = \begin{pmatrix} D_0 \\ \vdots \\ D_d \end{pmatrix}. \tag{24}$$

Step 3. Update the state x using the following searching iteration:

$$\begin{aligned} x^{k+1} &\triangleq x^k - \gamma\left(\Xi'\left(x^k\right)\right)^{+}\Xi\left(x^k\right) \\ &= x^k - \gamma\left(\Xi'\left(x^k\right)\right)^{+}\left[\Theta\left(x^k\right) - Y_d\right] \\ &= x^k - \gamma\left(\Theta'\left(x_0\right)\right)^{+}\left[\Theta\left(x^k\right) - Y_d\right], \end{aligned} \tag{25}$$

where superscript "+" means the pseudoinverse of a matrix; that is,

$$\begin{aligned} Q^{+} &= \left(Q^T Q\right)^{-1} Q^T, \\ \Xi\left(x\right) &= \Theta\left(x\right) - Y_d. \end{aligned} \tag{26}$$

Step 4. Go to Step 1 until the terminal conditions are satisfied.

If the system is nonautonomous, that is, it can be rewritten as following form,

$$\begin{aligned} \dot{x} &= f\left(x, u\right), \\ y &= h\left(x\right). \end{aligned} \tag{27}$$

We can first extend it into the following form:

$$\begin{aligned} \dot{x} &= f\left(x, u\right), \\ \dot{u} &= 0, \\ y &= h\left(x\right) \end{aligned} \tag{28}$$

and the new system can be rewritten as

$$\begin{aligned} \dot{\overline{x}} &= F\left(\overline{x}\right), \\ y &= H\left(\overline{x}\right), \end{aligned} \tag{29}$$

where

$$\overline{x} = \begin{bmatrix} x \\ u \end{bmatrix}. \tag{30}$$

Thus, the inverse computation can be conducted using the preceding algorithm.

As for the proposed quadrocopter system introduced in Section 2, this algorithm can be directly utilized through defining the following output vector:

$$Y \triangleq \left[p^T \ \dot{p}^T \ \ddot{p}^T \ \dddot{p}^T \ \left(p^{(t)}\right)^T \ \psi \ \dot{\psi} \ \ddot{\psi} \right]^T_{18\times 1}. \tag{31}$$

4. Evaluation of Circle Flying Maneuver

In this section, a so-called "Circle Maneuver," that is, the RFR system flies along a circle as shown in the following equation (32) and Figure 2 [10], of the Quadrocopter is taken as an example to show the validity of the proposed algorithm. Consider

$$\begin{aligned} \psi_d &= \pi + \frac{t}{T} \times 2\pi, \\ x_d &= 50 \sin\left(\frac{t}{T} \times 2\pi\right), \\ y_d &= 50 \cos\left(\frac{t}{T} \times 2\pi\right), \\ z_d &= 0, \end{aligned} \tag{32}$$

where T is the period of the circle maneuver.

The parameters of the RFR system are as follows:

$$m = 9.5\,\text{kg}$$
$$J = \begin{bmatrix} 0.1364 & 0 & 0 \\ 0 & 0.5782 & 0 \\ 0 & 0 & 0.6306 \end{bmatrix} \text{N} \cdot \text{m}. \tag{33}$$

The trajectory feasibility is decided by some constrains on both inputs and states. Here the following constrains are considered.

Input Constrains. Consider

$$\begin{aligned} T_M &\in [49.4, 190.8]\,\text{N}, \qquad M_x \in [-25.5, 25.5]\,\text{N} \cdot \text{m}, \\ M_y &\in [-40.2, 40.2]\,\text{N} \cdot \text{m}, \qquad M_z \in [-34.6, 34.6]\,\text{N} \cdot \text{m}. \end{aligned} \tag{34}$$

Angular Velocity Constrains. Consider

$$p, q \in [-0.4, 0.4]\,\text{rad/s}, \qquad r \in [-1.45, 1.45]\,\text{rad/s}. \tag{35}$$

Attitude Constrains. Consider

$$\theta, \phi \in [-0.5, 0.5]\,\text{rad}. \tag{36}$$

In this paper, the software toolbox LIEDRIVERS [17] based on ADOL-C is used as the basic AD algorithm, and the optimization searching is implemented using the golden sector algorithm, instead of the direct derivative-based algorithm as in Section 4. During the simulations, we test the circle maneuvers with different period T, and the results are given in Figures 3, 4, and 5.

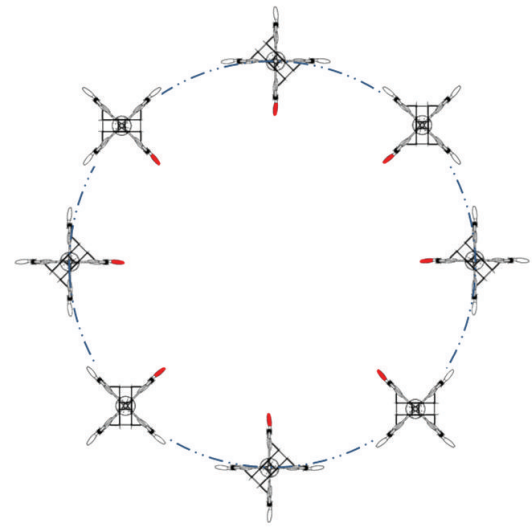


Figure 2: Circle maneuver trajectory.

From these figures, we can obtain the following results.

(1) For input constraints (Figures 3 and 5), only the maneuvering trajectories with $T = 10$ s are infeasible because the required dragging force is almost 210 N, larger than the maximum permitted value of 190.8 N, while the other inputs, that is, the three moments, are all much smaller than the maximum values.

(2) For angular velocity constrains (Figure 5), again, only the maneuvering trajectories with $T = 10$ s are infeasible, and the maneuvering trajectories with $T = 12.5$ s are *dangerous* since the maximum angular velocity approaches the maximum value of the permitted angular velocity.

(3) Based on the attitude constrains (Figure 4), the maneuvering trajectories with $T = 14.3$ s, 12.5 s, and 10 s are all infeasible.

Finally, from the preceding results, it can be seen that the circle maneuvers are really a highly dynamical maneuver, and the quantitative analysis prior to the flight test is useful and necessary to evaluate whether or not a maneuver trajectory is feasible or even dangerous.

5. Conclusions

In this paper, the flight maneuvering trajectories evaluation problem is researched, and the algorithm differentiation (AD) is used to realize the inverse computation of the rotor-flying robot (RFR) system. This paper starts from constructing nonlinear dynamical model of an RFR system. Then, we show that the RFR system model is flat taking translational positions and yaw angle as flat outputs. After that, the scheme of AD based inverse computation is introduced, that is,

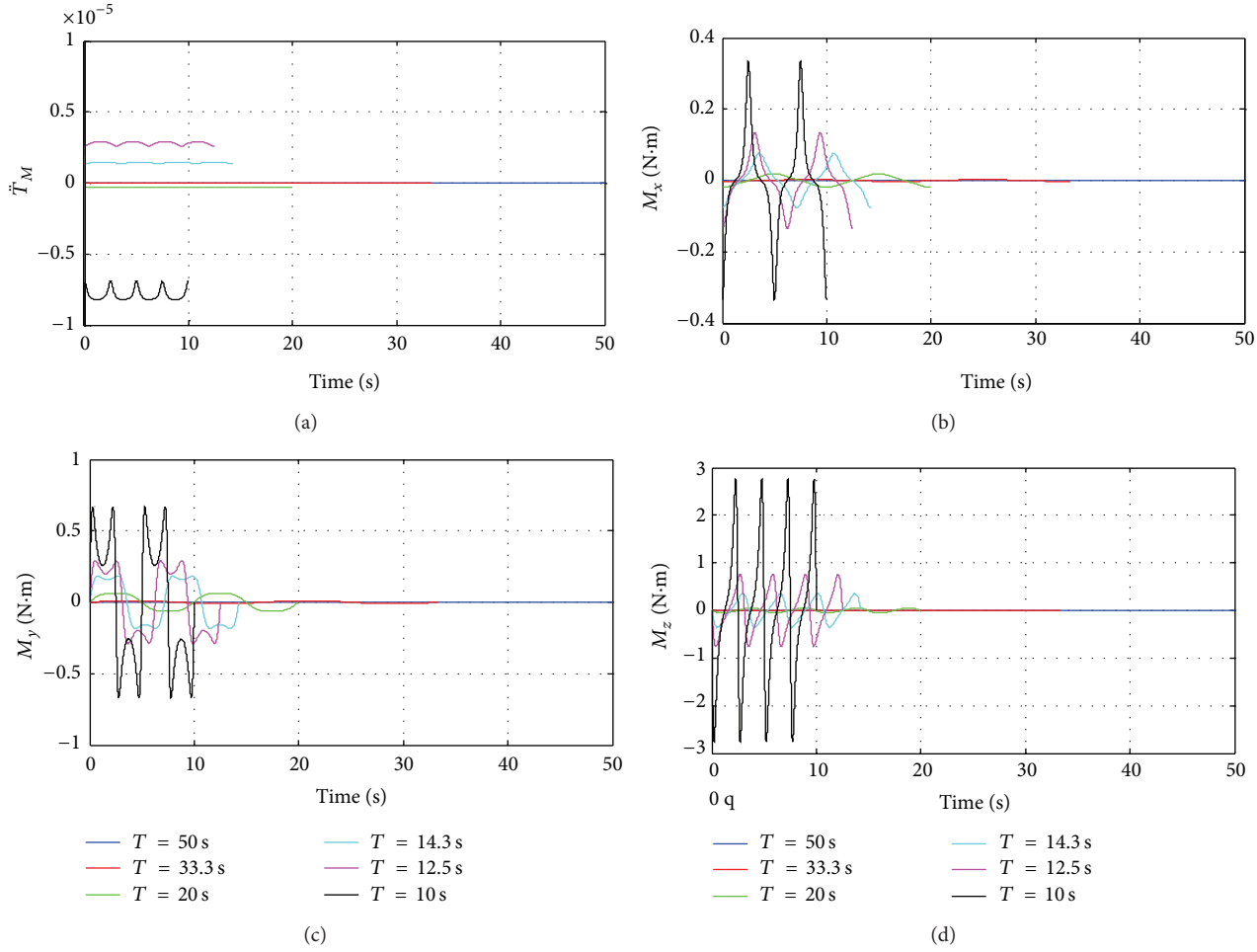


Figure 3: Control input: $\ddot{T}_M$ (a); M_x (b); M_y (c); M_z (d).

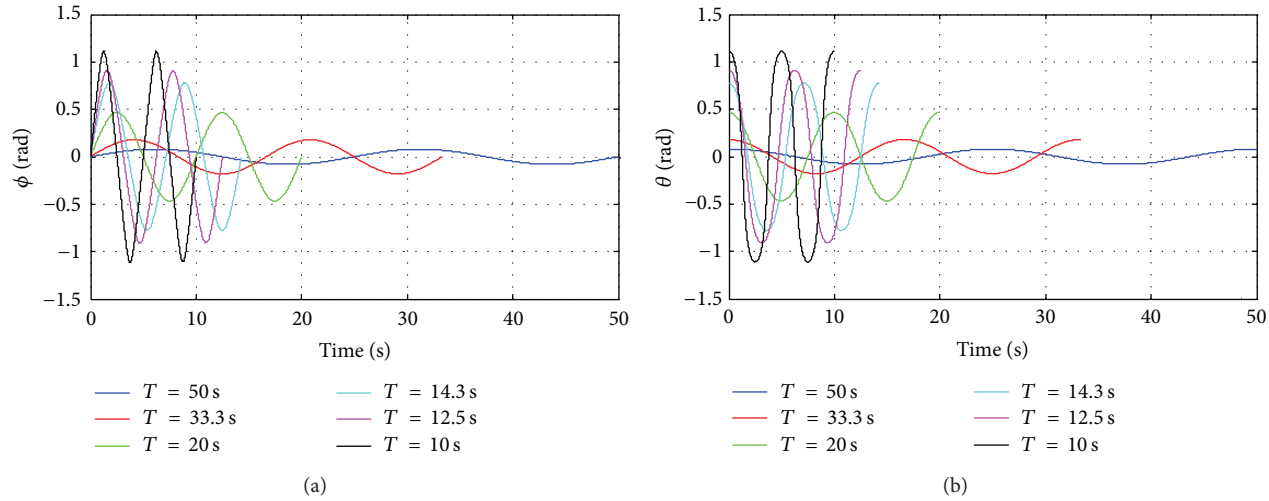


Figure 4: State: ϕ (a); θ (b).

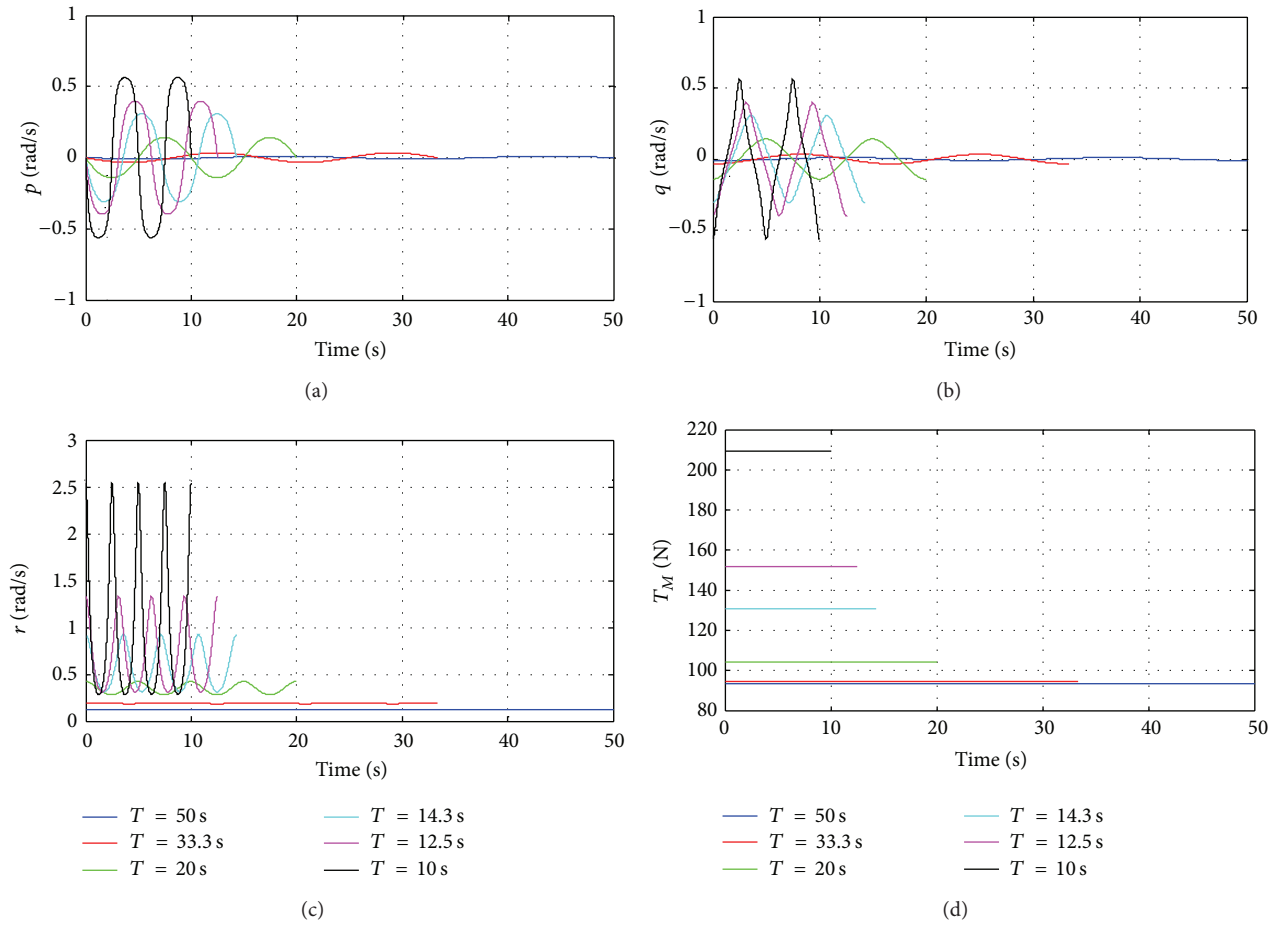


FIGURE 5: State: p (a); q (b); r (c); T_M (d).

computing the internal states and inputs based on the desired flat outputs. Finally, with the proposed scheme, a typical flight maneuver, that is, the circle maneuver, is analyzed.

The advantages of our proposed algorithm are as follows: (1) only desired positions and their derivatives, which is the case for many planning algorithm, and the feasibility of the trajectories (including inner states and inputs) can be evaluated; (2) inverse computation can be very accurate because AD is a numerical algorithm that can precisely obtain the derivative of a nonlinear function. Thus, an accurate pointwise numerical feasibility analysis of planned trajectories becomes possible.

The method proposed in this paper can be used to not only evaluate the feasibility and validity of the maneuver flying trajectories, but also realize the tracking control algorithm, where the computed internal states and inputs are useful for attenuating the online computational burden of the tracking controller and thus improving the closed loop performance. This will be one of our future's works.

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 present their thanks to Professor Klaus Röbenack at Technique University of Dresden (Germany) for the related discussion and suggestions on AD algorithm. This work is supported by the National Natural Science Foundation of China (Grants no. 61035005 and no. 61203340).

References

[1] Y. Q. He and J. D. Han, "Acceleration-feedback-enhanced robust control of an unmanned helicopter," *Journal of Guidance, Control, and Dynamics*, vol. 33, no. 4, pp. 1236–1250, 2010.

[2] J. H. Gillula, G. M. Hoffmann, M. P. Vitus, and C. J. Tomlin, "Applications of hybrid reachability analysis to robotic aerial vehicles," *International Journal of Robotics Research*, vol. 30, no. 3, pp. 335–354, 2011.

[3] F. Andert, F. Adolf, L. Goormann, and J. Dittrich, "Mapping and path planning in complex environments: an obstacle avoidance approach for an unmanned helicopter," in *Proceedings of the IE E E International Conference on Robotics and Automation*, pp. 745–750, Shanghai, China, May 2011.

[4] G. Flores, S. T. Zhou, R. Lozano, and P. Castillo, "A vision and GPS-based real time trajectory planning for a MAV in unknown

and low-sunlight environments," *Journal of Intelligent Robotic Systems*, vol. 74, pp. 59–67, 2014.

[5] P. Bristeau, E. Dorveaux, D. Vissière, and N. Petit, "Hardware and software architecture for state estimation on an experimental low-cost small-scaled helicopter," *Control Engineering Practice*, vol. 18, no. 7, pp. 733–746, 2010.

[6] A. I. Mourikis, N. Trawny, S. I. Roumeliotis, A. E. Johnson, A. Ansar, and L. Matthies, "Vision-aided inertial navigation for spacecraft entry, descent, and landing," *IEEE Transactions on Robotics*, vol. 25, no. 2, pp. 264–280, 2009.

[7] M. Bangura and R. Mahony, "Nonlinear dynamic modeling for high performance control of a quadrotor," in *Proceedings of the Australasian Conference on Robotics and Automation (ACRA '12)*, pp. 1–10, Wellington, New Zealand, December 2012.

[8] G. M. Hoffmann, H. Huang, S. L. Waslander, and C. J. Tomlin, "Precision flight control for a multi-vehicle quadrotor helicopter testbed," *Control Engineering Practice*, vol. 19, no. 9, pp. 1023–1036, 2011.

[9] M. W. Mueller, M. Hehn, and R. D'Andrea, "A computationally efficient algorithm for state-to-state quadrocopter trajectory generation and feasibility verification," in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS '13)*, pp. 3480–3486, Tokyo, Japan, 2013.

[10] D. Song, J. Han, and G. Liu, "Active model-based predictive control and experimental investigation on unmanned helicopters in full flight envelope," *IEEE Transactions on Control Systems Technology*, vol. 21, no. 4, pp. 1502–1509, 2013.

[11] O. Purwin and R. D'Andrea, "Performing aggressive maneuvers using iterative learning control," in *Proceedings of the IEEE International Conference on Robotics and Automation*, pp. 1731–1736, Kobe, Japan, May 2009.

[12] S. Lupashin, A. Schöllig, M. Sherback, and R. D'Andrea, "A simple learning strategy for high-speed quadrocopter multi-flips," in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA '10)*, pp. 1642–1648, Anchorage, Alaska, USA, May 2010.

[13] K. Röbenack and O. Vogel, "Computation of state and input trajectories for flat systems using automatic differentiation," *Automatica*, vol. 40, no. 3, pp. 459–464, 2004.

[14] G. Andreas and W. Andrea, *Evaluating Derivatives: Principles and Techniques of Algorithm Differentiation*, Society for Industrial and Applied Mathematics, Philadelphia, Pa, USA, 2nd edition, 2008.

[15] B. Christianson, "Reverse accumulation and accurate rounding error estimates for Taylor series coefficients," *Optimization Methods and Software*, vol. 1, no. 1, pp. 81–94, 1992.

[16] K. Röbenack, "Automatic differentiation and nonlinear controller design by exact linearization," *Future Generation Computer Systems*, vol. 21, no. 8, pp. 1372–1379, 2005.

[17] K. Röbenack, J. Winkler, and S. Q. Wang, "LIEDRIVERS—a toolbox for the efficient computation of Lie derivatives based on the object-oriented algorithmic differentiation package ADOL-C," in *Proceedings of the 4th International Workshop on Equation-Based Object-Oriented Modeling Languages and Tools*, ETH Zurich, Zurich, Switzerland, 2011.

Hindawi Publishing Corporation
Mathematical Problems in Engineering
Volume 2014, Article ID 206926, 12 pages
http://dx.doi.org/10.1155/2014/206926

Research Article

Bound Alternative Direction Optimization for Image Deblurring

Xiangrong Zeng[1,2]

[1] *College of Information System and Management, National University of Defense Technology, Changsha 410073, China*
[2] *Instituto Superior Tecnico, Universidade de Lisboa, 1049001 Lisboa, Portugal*

Correspondence should be addressed to Xiangrong Zeng; zengxrong@gmail.com

Received 8 March 2014; Revised 19 July 2014; Accepted 20 July 2014; Published 13 August 2014

Academic Editor: Fatih Yaman

This paper proposes a new method, *bound alternative direction method* (BADM), to address the ℓ_p ($p \in (0,1)$) minimization problems in image deblurring. The approach is to first obtain a bound unconstrained problem through bounding the ℓ_p regularizer by a novel majorizer and then, based on a variable splitting, to reformulate the bound unconstrained problem into a constrained one, which is then addressed via an augmented Lagrangian method. The proposed algorithm actually combines the reweighted ℓ_1 minimization method and the *alternating direction method of multiples* (ADMM) such that it succeeds in extending the application of ADMM to ℓ_p minimization problems. The conducted experimental studies demonstrate the superiority of the proposed algorithm for the synthesis ℓ_p minimization over the state-of-the-art algorithms for the synthesis ℓ_1 minimization on image deblurring.

1. Introduction

The mission of image deblurring is to restore an original image $\mathbf{x}$ from noisy blurred observation $\mathbf{y} \in \mathbb{R}^m$ modeled as

$$\mathbf{y} = \mathbf{A}\mathbf{x} + \mathbf{n}, \tag{1}$$

where $\mathbf{x} \in \mathbb{R}^n$ (stacking an $M \times N$-image into an ($n = MN$)-vector in lexicographic order), $\mathbf{A} \in \mathbb{R}^{m\times n}$ is the matrix representation of a convolution operator, and $\mathbf{n} \in \mathbb{R}^m$ is Gaussian white noise. This imaging inverse problem has recently inspired a considerable amount of research ([1–11] and further references therein) in which the optimization problems fall into two varieties: synthesis formulation and analysis formulation [12] which are detailed below.

1.1. Formulations and Algorithms. In the synthesis formulation, the unknown image $\mathbf{x}$ is represented as $\mathbf{x} = \mathbf{W}\mathbf{s}$, where $\mathbf{W}$ is a wavelet frame or a redundant dictionary and $\mathbf{s}$ are the sparse coefficients estimated usually via one sparsity-promoting regularizer, yielding

$$\mathbf{s}^* = \arg\min_{\mathbf{s}} \frac{1}{2}\left\|\mathbf{A}\mathbf{W}\mathbf{s} - \mathbf{y}\right\|_2^2 + \lambda\varphi(\mathbf{s}), \tag{2}$$

where λ is the positive regularization parameter and φ is the regularizer, typically the ℓ_1 norm. The deblurred image is then obtained by $\mathbf{x}^* = \mathbf{W}\mathbf{s}^*$.

In the analysis formulation, as opposed to (2), it minimizes the cost function with respect to $\mathbf{x}$:

$$\mathbf{x}^* = \arg\min_{\mathbf{x}} \frac{1}{2}\left\|\mathbf{A}\mathbf{x} - \mathbf{y}\right\|_2^2 + \lambda\varphi(\mathbf{T}\mathbf{x}), \tag{3}$$

where $\mathbf{T}$ is a sparsifying transform (such as wavelet or finite differences) and $\varphi(\mathbf{T}\mathbf{x})$ analyzes the image $\mathbf{x}$ itself against the coefficients $\mathbf{s}$ that $\varphi(\mathbf{s})$ in (2) works on. If $\mathbf{T} = \mathbf{W}$ is a wavelet frame, then usually $\varphi(\mathbf{T}\mathbf{x}) = \|\mathbf{T}\mathbf{x}\|_1$ where $\|\mathbf{v}\|_1 = \sum_i v_i$; if $\mathbf{T}$ is a matrix representing finite differences at horizontal and vertical directions, then $\varphi(\mathbf{T}\mathbf{x})$ is the discrete anisotropic or isotropic total variation [8, 9].

In the past several decades, a lot of algorithms have been proposed to solve (2) and (3). Among these, the iterative shrinkage/thresholding (IST, [2]) algorithm can be considered as the standard one. However, IST tends to be slow in particular when $\mathbf{A}$ in (1) is poorly conditioned. To overcome this problem, several fast variants of IST have been proposed. They are TwIST [13], FISTA [14], and SpaRSA [15]. TwIST is actually a two-step version of IST in which each iterate depends on the two previous iterates, rather than

only on the previous one (as in IST); FISTA is a nonsmooth variant of Nesterov's optimal gradient-based algorithm for smooth convex problems [16, 17]; and SpaRSA adopted more aggressive choices of step size in each iteration. These three algorithms, which only use the gradient information of the data-fitting term, have been shown to clearly outperform IST. Some other efficient algorithms, using the second-order information of the data-fitting term, have also been proposed. A noble representative is the so-called split augmented Lagrangian shrinkage algorithm (SALSA) [10, 11], which was found to be faster than the previous FISTA, TwIST, and SpaRSA when the matrix inversion $(\mathbf{A}^T\mathbf{A} + \mu\mathbf{I})^{-1}$ (where μ is a positive parameter) can be efficiently computed (*e.g.,* if $\mathbf{A}$ is a circulant matrix, then the matrix inversion can be efficiently calculated by FFT). Actually, SALSA is an instance of *alternating direction method of multipliers* (ADMM) [18, 19] which has a close relationship [20] with the Bregman iterations [21–24], amongst which, the split Bregman method (SBM) [22] has been recently frequently applied to handle imaging inverse problems [25–27].

1.2. ℓ_p Minimization. Convergences of above algorithms are guaranteed, benefiting from the fact that the regularizer φ in (2) and (3) is usually convex. In this paper, a nonconvex (also nonsmooth) regularizer, that is, the ℓ_p ($p \in (0, 1)$) norm, is adopted since using the ℓ_p norm is able to find sparser solution than using the ℓ_1 norm which was demonstrated in many studies [28–32]. Thus, there comes the ℓ_p minimization problem of the synthesis formulation:

$$\min_{\mathbf{s}} \frac{1}{2}\left\|\mathbf{AWs} - \mathbf{y}\right\|_2^2 + \lambda\|\mathbf{s}\|_p^p \tag{4}$$

and of the analysis formulation:

$$\min_{\mathbf{x}} \frac{1}{2}\left\|\mathbf{Ax} - \mathbf{y}\right\|_2^2 + \lambda\|\mathbf{Tx}\|_p^p, \tag{5}$$

where $\|\mathbf{v}\|_p^p = \sum_i |v_i|^p$.

Next, a brief survey on the literature of ℓ_p minimization is given. Recently, remarkable attention has been paid to the ℓ_p minimization problem:

$$\min_{\mathbf{x}} \|\mathbf{x}\|_p^p : \ \text{s.t. } \mathbf{Ax} = \mathbf{y}, \tag{6}$$

where $p \in (0, 1)$. Some sufficient conditions of (6) have been established in [29, 33–35]. Many efficient reweighted minimization methods address (6) through iteratively solving

$$\mathbf{x}^{k+1} = \arg\min_{\mathbf{x}} f(\mathbf{x}) + \lambda\psi^{k+1}(\mathbf{x}), \tag{7}$$

where $\mathbf{x}^{k+1}$ is the estimated signal at the $(k + 1)$th iteration; f is the data-fitting term which preserves the consistency; for instance, $f(\mathbf{x}) = (1/2)\|\mathbf{Ax} - \mathbf{y}\|_2^2$ for the Gaussian noise; $\psi^{k+1}(\mathbf{x})$ is the regularization term at the $(k + 1)$th iteration, which is aiming to encourage the desirable properties of the target $\mathbf{x}$ such as sparsity. Most of reweighted minimization methods (see [2, 28, 30, 31, 34, 36, 37] and many references therein) focus on the reweighted ℓ_1 (IRL1) and the reweighted ℓ_2 (IRL2) minimization, and their corresponding regularizers are $\psi^{k+1}(\mathbf{x}) = \sum_i w_i^{k+1}|x_i|$ and $\sum_i w_i^{k+1}|x_i|^2$, respectively, where x_i is the ith component of $\mathbf{x}$ and w_i^{k+1} is the weight of x_i at the kth iteration and it is a function of the previous estimated component x_i^k. In IRL1, usually, $w_i^k = (|x_i^k| + \epsilon)^{q-1}$, while $w_i^k = (|x_i^k|^2 + \epsilon)^{q/2-1}$ in IRL2, where $q \in (0, 1]$. Notice that q is always set to 1 (e.g., IRL1: [28] and IRL2: [2]) or p (e.g., IRL1: [34, 36, 38] and IRL2: [30, 31, 38]), even 1/2 [32]. It is worth noting that above weights are separable, even though there also exist some reweighted minimization methods [39, 40] with inseparable weights.

1.3. Proposed Approach. In this paper, the ℓ_p minimization problem will be used for image deblurring, but the resulting problem cannot be efficiently solved by above ℓ_p minimization methods stated in Section 1.2, since they usually adopt slow iterations (such as the generic IST algorithm), and in image deblurring, the matrix-vector products are always computationally expensive. Considering this, the ADMM is used in this paper to tackle the resulting ℓ_p minimization problem because as stated in Section 1.1, the image deblurring problems can be efficiently handled by the ADMM. The proposed approach is to first bound it via a variant of the majorizer proposed in [4, 7], obtaining a bound unconstrained one; then reformulate it into a constrained problem via the technique of variable splitting [41, 42]; and lastly attack this resulting constrained problem using an augmented Lagrangian method (ALM) [43]. If the majorizer that is proposed in [4, 7] is used to bound the original ℓ_p minimization unconstrained problem, then the components of the estimate in the iterations would be stuck at zero forever if they became zero, which very likely prevents convergence to a minimizer. To overcome this problem, in this paper, a variant of this majorizer is proposed by adding a small positive parameter that shrinks in each iteration. The obtained bound unconstrained problem is reformulated into a constrained one through variable splitting which splits the original variable into a pair of variables, serving as the arguments of the data-fitting term and the ℓ_p regularizer, respectively, under the constraint that these two variables have to be equal. This resulting constrained problem is then attacked by an ALM, obtaining a Lagrangian function with two variables resulting from variable splitting. Next, the Lagrangian function is alternatively solved with respect to the two variables, leading to the method called *bound alternative direction method* (BADM), which is equivalent to the combination of the reweighted ℓ_1 minimization method and the ADMM and is able to extend the application of ADMM to the ℓ_p minimization problems.

The proposed BADM has only $\mathcal{O}(n \log n)$ cost in each iteration in solving the synthesis ℓ_p formulation with a normalized Parseval frame. Experiments on a set of benchmark problems show that the BADM for (4) is favorably competitive with the state-of-the-art algorithms FISTA [14], SALSA [11], and SBM [22] for (2).

1.4. Terminology and Notation. In this section, some useful elements of convex analysis will be given. Let $\mathscr{H}$ be a real Hilbert space equipped with the inner product $\langle\cdot,\cdot\rangle$ and associated norm $\|\cdot\|$. Let $f:\mathscr{H}\to[-\infty,+\infty]$ be a function and let $\Gamma(\mathbf{x})$ be the class of all lower semicontinuous convex functions that are not equal to $+\infty$ everywhere and are never equal to $-\infty$. The proximity operator of f is defined as

$$\mathrm{Prox}_{\tau f}(\mathbf{v}) = \arg\min_{\mathbf{x}}\left(\tau f(\mathbf{x}) + \frac{1}{2}\|\mathbf{x}-\mathbf{v}\|^2\right), \tag{8}$$

where τ is positive. If $f\in\Gamma(\mathbf{x})$, then $\mathrm{Prox}_{\tau f}$ is unique [44]; if $f(\mathbf{x}) = \iota_C(\mathbf{x})$, the indicator function of a nonempty closed convex set C, then $\mathrm{Prox}_{\tau f}(\mathbf{x})$ becomes the projection of $\mathbf{v}$ onto C, and in this sense, (8) is therefore a generalization of projection operator; and if f is the ℓ_1 norm, then (8) becomes a well-known soft thresholding:

$$\mathrm{soft}(\mathbf{v},\tau) = \mathrm{sign}(\mathbf{v})\odot\max\{|\mathbf{v}|-\tau,0\}, \tag{9}$$

where $\odot$ is the componentwise multiplication between two vectors of the same dimension and $\mathrm{sign}(\cdot)$ is the sign function.

2. Bound Alternative Direction Optimization

Consider a ℓ_p regularized unconstrained model

$$\min_{\mathbf{x}} F(\mathbf{x}) := f(\mathbf{x}) + \lambda\|\mathbf{x}\|_p^p, \tag{10}$$

where $p\in(0,1)$ and $f:\mathbb{R}^n\to\mathbb{R}$ is a smooth function with L-Lipschitz-continuous gradient and is bounded below. It is unsuitable to directly apply the existing proximal splitting algorithms (such as IST, FISTA, TwIST, and SpaRSA discussed in the section of Introduction) to solve (10), since, for $p\in(0,1)$, the nonconvex nature of $\|\mathbf{x}\|_p^p$ blocks the use of the proximity operator (see (8)), which is well defined only on the functions which belong to $\Gamma(\mathbf{x})$. To overcome this problem, a bound optimization approach will be considered in this paper.

2.1. Bound Optimization

2.1.1. Bound $F(\mathbf{x})$ via the Majorizer Proposed in [7]. Since $\|\mathbf{x}\|_p^p$ $(p\in(0,1))\notin\Gamma(\mathbf{x})$, it is reasonable to bound $F(\mathbf{x})$ through

$$\widehat{G}(\mathbf{x},\widetilde{\mathbf{x}}) = f(\mathbf{x}) + \widehat{\Phi}(\mathbf{x},\widetilde{\mathbf{x}}), \tag{11}$$

where $\widehat{\Phi}(\mathbf{x},\widetilde{\mathbf{x}}) = \lambda\sum_i\widehat{\phi}(x_i,\widetilde{x}_i)$ with

$$\widehat{\phi}(x,\widetilde{x}) = \begin{cases}\widehat{\Omega}(\widetilde{x})|x| + (1-p)|\widetilde{x}|^p, & \text{if } \widetilde{x}\neq 0,\\ +\infty, & \text{if } \widetilde{x}=0,\ x\neq 0,\\ 0, & \text{if } \widetilde{x}=0,\ x=0,\end{cases} \tag{12}$$

where $\widehat{\Omega}(\widetilde{x}) = p|\widetilde{x}|^{p-1}$. It is easy to see that $F(\mathbf{x})\le G(\mathbf{x},\widetilde{\mathbf{x}})$ with equality for $\mathbf{x}=\widetilde{\mathbf{x}}$, since $\lambda\|\mathbf{x}\|_p^p\le\widehat{\Phi}(\mathbf{x},\widetilde{\mathbf{x}})$ with equality for $\mathbf{x}=\widetilde{\mathbf{x}}$. One benefit from the above bound is that the following lemma holds.

Lemma 1. *Given $\widetilde{\mathbf{x}}$, $\widehat{\Phi}(\mathbf{x},\widetilde{\mathbf{x}})$ belongs to $\Gamma(\mathbf{x})$ with respect to $\mathbf{x}$.*

Proof. Given $\widetilde{x}_i$ for $i=1,\ldots,n$, if $\widetilde{x}_i\neq 0$, then $\widehat{\phi}(x_i,\widetilde{x}_i)$ is an affine function of $|x_i|$ and thus $\widehat{\phi}(x_i,\widetilde{x}_i)\in\Gamma(x_i)$, since $|x_i|\in\Gamma(x_i)$; if $\widetilde{x}_i=0$, then $\widehat{\phi}(x_i,\widetilde{x}_i)$ equals 0 if $x_i=0$ and $+\infty$ otherwise, leading to $\widehat{\phi}(x_i,\widetilde{x}_i)\in\Gamma(x_i)$. Therefore, $\widehat{\Phi}(\mathbf{x},\widetilde{\mathbf{x}}) = \lambda\sum_i\widehat{\phi}(x_i,\widetilde{x}_i)\in\Gamma(\mathbf{x})$. □

Therefore, the proximity operator of $\widehat{\Phi}(\mathbf{x},\widetilde{\mathbf{x}})$ given $\widetilde{\mathbf{x}}$ is obtained by

$$\begin{aligned}\mathrm{Prox}_{\widehat{\Phi}(\mathbf{x},\widetilde{\mathbf{x}})}(\mathbf{v}) &= \arg\min_{\mathbf{x}}\left(\widehat{\Phi}(\mathbf{x},\widetilde{\mathbf{x}}) + \frac{1}{2}\|\mathbf{x}-\mathbf{v}\|^2\right)\\ &= \mathrm{soft}(\mathbf{v},\widehat{\boldsymbol{\omega}}(\widetilde{\mathbf{x}})),\end{aligned} \tag{13}$$

where $\widehat{\boldsymbol{\omega}}(\widetilde{\mathbf{x}}) = [w_1(\widetilde{x}_1),\ldots,w_n(\widetilde{x}_n)]^T$ with

$$w_i(\widetilde{x}_i) = \begin{cases}\lambda\widehat{\Omega}(\widetilde{x}_i), & \text{if } \widetilde{x}_i\neq 0,\\ +\infty, & \text{if } \widetilde{x}_i=0.\end{cases} \tag{14}$$

Notice that $\mathrm{soft}(x,+\infty)=0$ for any x.

From above, a closed-form solution of the proximity operator of $\widehat{\Phi}(\mathbf{x},\widetilde{\mathbf{x}})$ can be obtained after the bound operation. However, in many iteratively minimization algorithms discussed in the section of Introduction, the proximity operator of the regularization term is commonly used. For $\widehat{\Phi}(\mathbf{x},\widetilde{\mathbf{x}})$, a nature way is to set $\widetilde{\mathbf{x}}$ as the previous estimate $\mathbf{x}^k$ and obtain the current estimate $\mathbf{x}^{k+1}$ by computing the proximity operator of $\widehat{\Phi}(\mathbf{x},\mathbf{x}^k)$; that is,

$$\mathbf{x}^{k+1} = \mathrm{Prox}_{\widehat{\Phi}(\mathbf{x},\mathbf{x}^k)}(\boldsymbol{\nu}^k) = \mathrm{soft}\left(\boldsymbol{\nu}^k,\widehat{\boldsymbol{\omega}}(\mathbf{x}^k)\right), \tag{15}$$

where $\boldsymbol{\nu}^k$ is an iteratively temporary variable which differs in each algorithm. According to (13) with (14), if a component of $\mathbf{x}^k$ becomes zero forever, then this component of $\mathbf{x}^{k+1}$ is set to zero and will be stuck at zero forever, which may prevent convergence to a local minimizer, letting alone a global one. To overcome this shortcoming, a new bound method is proposed below.

2.1.2. Proposed Bound Method. A method is presented that bounds $F(\mathbf{x})$ through

$$G_\epsilon(\mathbf{x},\widetilde{\mathbf{x}}) = f(\mathbf{x}) + \Phi_\epsilon(\mathbf{x},\widetilde{\mathbf{x}}), \tag{16}$$

where $\Phi_\epsilon(\mathbf{x},\widetilde{\mathbf{x}}) = \lambda\sum_i\phi_\epsilon(x_i,\widetilde{x}_i)$ with

$$\phi_\epsilon(x,\widetilde{x}) = \Omega_\epsilon(\widetilde{x})|x| + (1-p)|\widetilde{x}|^p, \tag{17}$$

where $\Omega_\epsilon(\widetilde{x}) = p(|\widetilde{x}|+\epsilon)^{p-1}$ and ϵ is a small positive parameter. It is clear that $F(\mathbf{x})\le G_\epsilon(\mathbf{x},\widetilde{\mathbf{x}})$ with equality for $\mathbf{x}=\widetilde{\mathbf{x}}=\mathbf{0}$ (where $\mathbf{0}$ is a vector composed of all zeros) or $\mathbf{x}=\widetilde{\mathbf{x}}$ with $\epsilon=0$. $\Phi_\epsilon(\mathbf{x},\widetilde{\mathbf{x}})$ has the following property.

Lemma 2. *$\Phi_\epsilon(\mathbf{x},\widetilde{\mathbf{x}})$ belongs to $\Gamma(\mathbf{x})$ with respect to $\mathbf{x}$.*

Proof. For any $\widetilde{x}_i$, $i=1,\ldots,n$, $\Omega_\epsilon(\widetilde{x}_i)>0$; thus $\phi_\epsilon(x_i,\widetilde{x}_i)$ is an affine function of $|x_i|$ and thus $\phi_\epsilon(x_i,\widetilde{x}_i)\in\Gamma(x_i)$. Therefore, $\Phi_\epsilon(\mathbf{x},\widetilde{\mathbf{x}}) = \lambda\sum_i\phi_\epsilon(x_i,\widetilde{x}_i)\in\Gamma(\mathbf{x})$. □

(1) Set $k = 0$, and choose an arbitrary $\mathbf{x}^0$, $\epsilon^0 > 0$ and $\gamma > 1$.
(2) **repeat**
(3) $\quad \mathbf{x}^{k+1} = \arg\min_{\mathbf{x}} G_{\epsilon^k}(\mathbf{x}, \mathbf{x}^k)$
(4) $\quad \epsilon^{k+1} = \epsilon^k / \gamma$
(5) $\quad k \leftarrow k + 1$
(6) **until** some stopping criterion is satisfied.

Algorithm 1: IBO.

(1) Set $k = 0$, and choose $\beta > 0$, $\gamma > 1$, $\epsilon^0 > 0$, $\boldsymbol{\alpha}^0$ and an arbitrary $\mathbf{x}^0$.
(2) **repeat**
(3) $\quad (\mathbf{v}^{k+1}, \mathbf{x}^{k+1}) \in \arg\min_{\mathbf{v},\mathbf{x}} \mathscr{L}(\mathbf{v}, \mathbf{x}; \mathbf{x}^k, \boldsymbol{\alpha}^k, \beta, \epsilon^k)$
(4) $\quad \boldsymbol{\alpha}^{k+1} = \boldsymbol{\alpha}^k - \beta(\mathbf{v}^{k+1} - \mathbf{x}^{k+1})$
(5) $\quad \epsilon^{k+1} = \epsilon^k / \gamma$
(6) $\quad k \leftarrow k + 1$
(7) **until** some stopping criterion is satisfied

Algorithm 2: BALM.

(1) Set $k = 0$, and choose $\beta > 0$, $\gamma > 1$, $\epsilon^0 > 0$, $\mathbf{d}^0$ and an arbitrary $\mathbf{x}^0$.
(2) **repeat**
(3) $\quad (\mathbf{v}^{k+1}, \mathbf{x}^{k+1}) \in \arg\min_{\mathbf{v},\mathbf{x}} f(\mathbf{v}) + \Phi_{\epsilon^k}\left(\mathbf{x}, \mathbf{x}^k\right) + (\beta/2)\left\|\mathbf{v} - \mathbf{x} - \mathbf{d}^k\right\|_2^2$
(4) $\quad \mathbf{d}^{k+1} = \mathbf{d}^k - (\mathbf{v}^{k+1} - \mathbf{x}^{k+1})$
(5) $\quad \epsilon^{k+1} = \epsilon^k / \gamma$
(6) $\quad k \leftarrow k + 1$
(7) **until** some stopping criterion is satisfied.

Algorithm 3: Variant of BALM.

(1) Set $k = 0$, and choose $\beta > 0$, $\gamma > 1$, $\epsilon^0 > 0$, $\mathbf{d}^0$ and an arbitrary $\mathbf{x}^0$.
(2) $\quad$ **repeat**
(3) $\quad \mathbf{v}^{k+1} = \arg\min_{\mathbf{v}} f(\mathbf{v}) + (\beta/2)\left\|\mathbf{v} - \mathbf{x}^k - \mathbf{d}^k\right\|_2^2$
(4) $\quad \mathbf{x}^{k+1} = \arg\min_{\mathbf{x}} \Phi_{\epsilon^k}(\mathbf{x}, \mathbf{x}^k) + (\beta/2)\left\|\mathbf{v}^{k+1} - \mathbf{x} - \mathbf{d}^k\right\|_2^2$
(5) $\quad \mathbf{d}^{k+1} = \mathbf{d}^k - (\mathbf{v}^{k+1} - \mathbf{x}^{k+1})$
(6) $\quad \epsilon^{k+1} = \epsilon^k / \gamma$
(7) $\quad k \leftarrow k + 1$
(8) **until** some stopping criterion is satisfied.

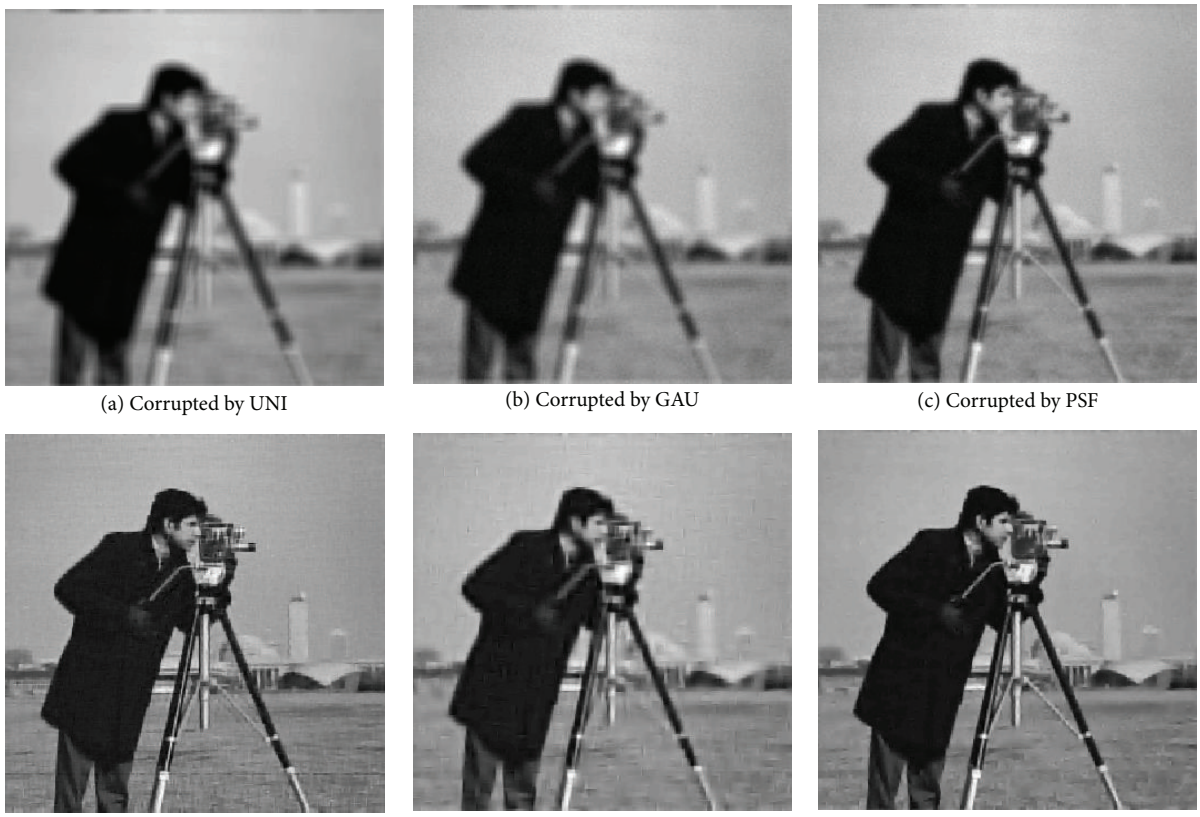
Algorithm 4: BADM.

(1) Set $k = 0$, and choose $\beta > 0$, $\gamma > 1$, $\epsilon^0 > 0$, $\mathbf{d}^0$ and an arbitrary $\mathbf{s}^0$.
(2) **repeat**
(3) $\quad \mathbf{v}^{k+1} = \left[(\mathbf{AW})^T \mathbf{AW} + \beta \mathbf{I}\right]^{-1} \left[(\mathbf{AW})^T \mathbf{y} + \beta\left(\mathbf{s}^k + \mathbf{d}^k\right)\right]$
(4) $\quad \mathbf{s}^{k+1} = \mathrm{soft}\left(\mathbf{v}^{k+1} - \mathbf{d}^k, \boldsymbol{\omega}_{\epsilon^k}\left(\mathbf{s}^k\right)/\beta\right)$
(5) $\quad \mathbf{d}^{k+1} = \mathbf{d}^k - (\mathbf{v}^{k+1} - \mathbf{s}^{k+1})$
(6) $\quad \epsilon^{k+1} = \epsilon^k / \gamma$
(7) $\quad k \leftarrow k + 1$
(8) **until** some stopping criterion is satisfied.

Algorithm 5: BADM for (4).

(a) Cameraman (b) Mandril (c) Lena

Figure 1: Test images.



(a) Corrupted by UNI (b) Corrupted by GAU (c) Corrupted by PSF

Deblurred from (a) Deblurred from (b) Deblurred from (c)

Figure 2: Deblurred Cameraman images by BADM.

Therefore, a closed-form solution of the proximity operator of $\Phi_\epsilon(\mathbf{x}, \widetilde{\mathbf{x}})$ can be obtained

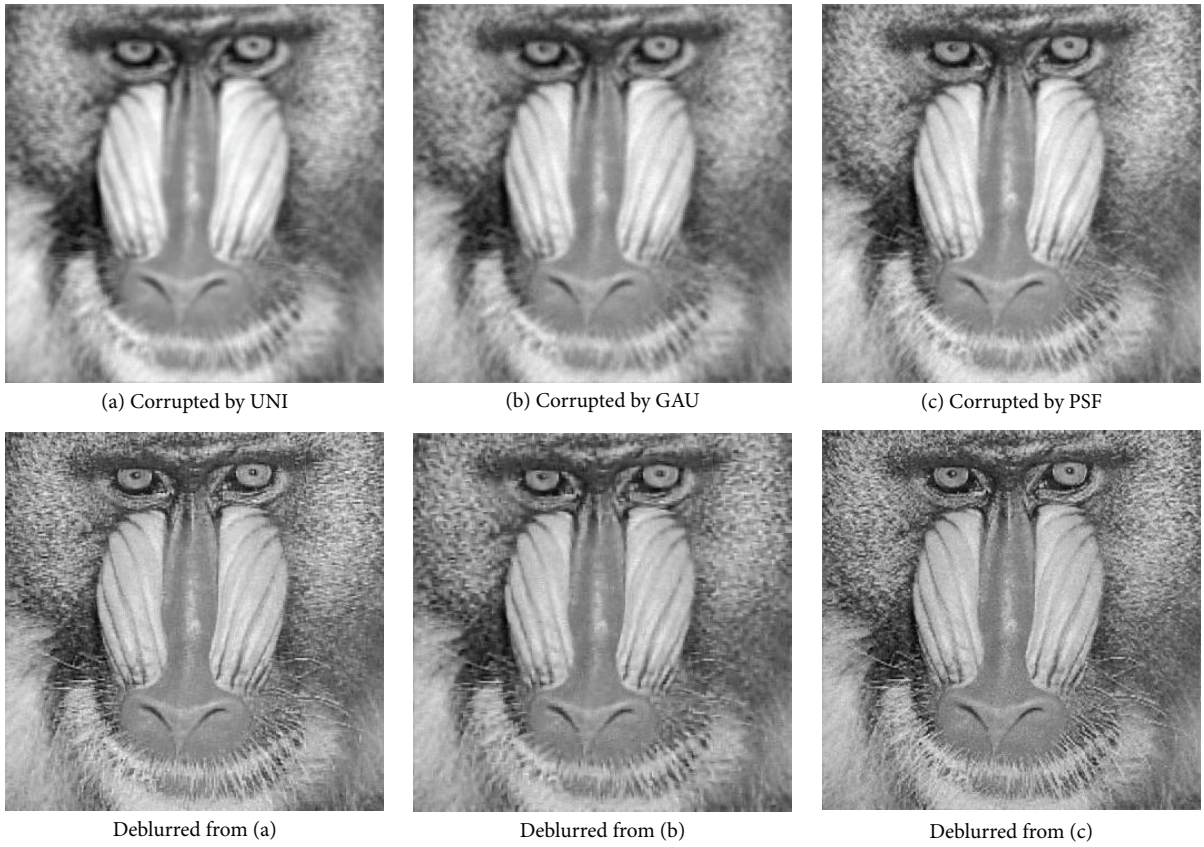
$$\begin{aligned} \mathrm{Prox}_{\Phi_\epsilon(\mathbf{x},\widetilde{\mathbf{x}})}(\mathbf{v}) &= \arg\min_{\mathbf{x}} \left(\Phi_\epsilon(\mathbf{x}, \widetilde{\mathbf{x}}) + \frac{1}{2}\|\mathbf{x} - \mathbf{v}\|^2 \right) \\ &= \mathrm{soft}\left(\mathbf{v}, \boldsymbol{\omega}_\epsilon(\widetilde{\mathbf{x}})\right), \end{aligned} \tag{18}$$

where $\boldsymbol{\omega}_\epsilon(\widetilde{\mathbf{x}}) = [\lambda\Omega_\epsilon(\widetilde{x}_1), \ldots, \lambda\Omega_\epsilon(\widetilde{x}_n)]^T$.

2.1.3. Iterative Bound Optimization (IBO). Now, it is ready to propose a framework of IBO as in Algorithm 1.

A key observation is that the sequence $\{\epsilon^k\}$, generated by the IBO, approaches to zero as $k \to +\infty$, such that (10) can be solved by the IBO which iteratively solves a sequence of problems:

$$\min_{\mathbf{x}} F_\epsilon(\mathbf{x}) := f(\mathbf{x}) + \lambda \sum_{i=1}^{n} \left(\left| x_i^k \right| + \epsilon^k \right)^p, \tag{19}$$



(a) Corrupted by UNI (b) Corrupted by GAU (c) Corrupted by PSF

Deblurred from (a) Deblurred from (b) Deblurred from (c)

Figure 3: Deblurred Mandril images by BADM.

where x_i^k is the ith component of $\mathbf{x}^k$. Any accumulation point of the sequence $\{\mathbf{x}^k\}$ generated by the IBO is a first-order stationary point of (19), which is guaranteed by the following theorem.

Theorem 3. *Let the sequence $\{\mathbf{x}^k\}$ be generated by above IBO and suppose that $\mathbf{x}^*$ is an accumulation point of $\{\mathbf{x}^k\}$; then $\mathbf{x}^*$ is a first-order stationary point of* (19).

Proof. IBO is actually a specific case of the reweighted ℓ_α (α = 1 or 2) method proposed in [45], such that this theorem is also a specific case of Theorem 3.1 in [45]. □

2.2. Bound Alternative Direction Method. Considering the unconstrained optimization problem that corresponds to Step 3 of IBO,

$$\min_{\mathbf{x}} G_\epsilon(\mathbf{x}, \widetilde{\mathbf{x}}) := f(\mathbf{x}) + \Phi_\epsilon(\mathbf{x}, \widetilde{\mathbf{x}}). \tag{20}$$

Using the technique of *variable splitting*, (20) becomes a constrained optimization problem:

$$\min_{\mathbf{v},\mathbf{x}} f(\mathbf{v}) + \Phi_\epsilon(\mathbf{x}, \widetilde{\mathbf{x}}) \quad \text{s.t. } \mathbf{v} = \mathbf{x}. \tag{21}$$

The rationale behind variable splitting is that it may be easier to solve (21) than it is to solve (20). The augmented Lagrangian function of (21) is

$$\begin{aligned}\mathscr{L}(\mathbf{v}, \mathbf{x}; \widetilde{\mathbf{x}}, \alpha, \beta, \epsilon) \\ = f(\mathbf{v}) + \Phi_\epsilon(\mathbf{x}, \widetilde{\mathbf{x}}) + \boldsymbol{\alpha}^T(\mathbf{v} - \mathbf{x}) + \frac{\beta}{2}\|\mathbf{v} - \mathbf{x}\|_2^2,\end{aligned} \tag{22}$$

where $\boldsymbol{\alpha}$ is the vector of Lagrangian multipliers and β is the penalty parameter. According to the augmented Lagrangian method (ALM) [43], (21) can be solved through repeating the following iterative process until some stopping criterion is satisfied: minimize (22) with respect to $\mathbf{v}$ and $\mathbf{x}$ fixing $\boldsymbol{\alpha}$ and setting $\widetilde{\mathbf{x}}$ to the previous estimate of $\mathbf{x}$ and then update $\boldsymbol{\alpha}$, yielding a variant of ALM called *bound ALM* (BALM) (Algorithm 2).

Notice that the terms added to $G_\epsilon(\mathbf{x}, \widetilde{\mathbf{x}})$ in the definition of $\mathscr{L}(\mathbf{v}, \mathbf{x}; \widetilde{\mathbf{x}}, \alpha, \beta, \epsilon)$ (see (22)) can be reformulated as a single quadratic term, leading to a variant of BALM (Algorithm 3).

In (Algorithm 3), $\mathbf{d}^k$ corresponds to $\boldsymbol{\alpha}^k/\beta$ that these two parameters are used in BALM. It is usually difficult to simultaneously obtain $\mathbf{v}^{k+1}$ and $\mathbf{x}^{k+1}$ in Step 3. To overcome this difficulty, the technique of *nonlinear block Gauss-Seidel* (NLBGS) [46] is naturally used, in which the objective function of Step 3 is solved by alternatively minimizing it with respect to $\mathbf{v}$ and $\mathbf{x}$, while keeping the other variable fixed,

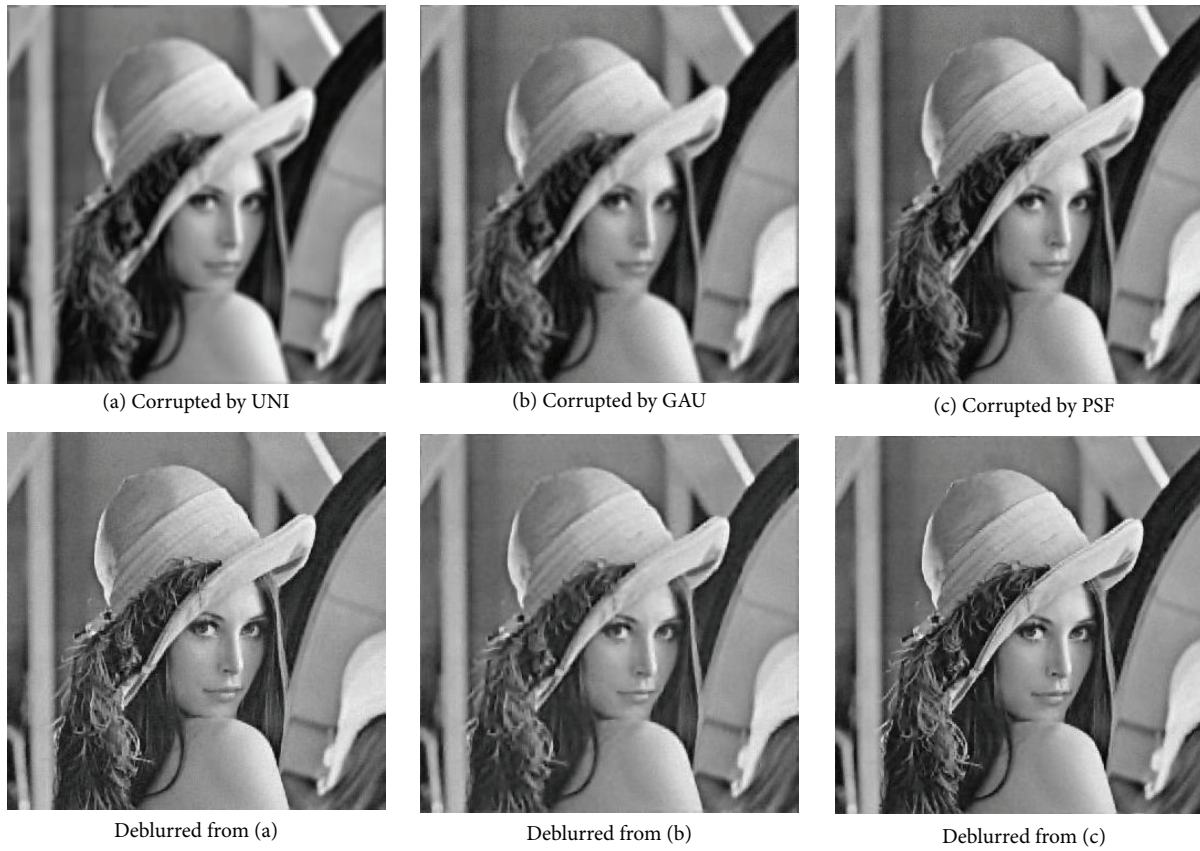


Figure 4: Deblurred Lena images by BADM.

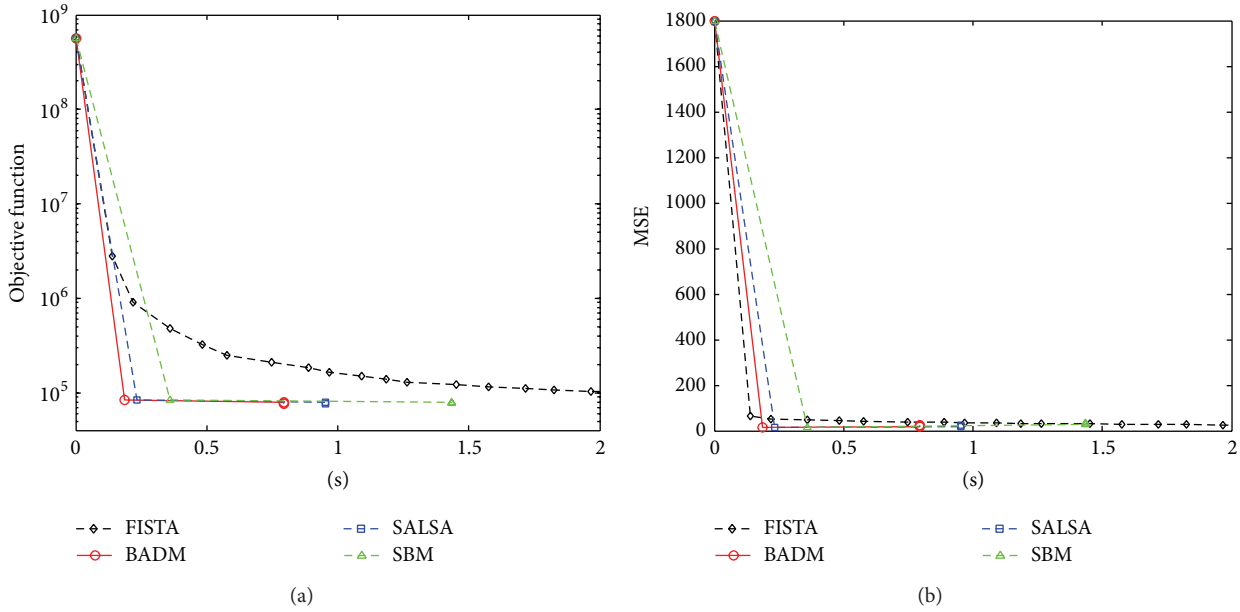


Figure 5: Evolutions in experiment (I) (Cameraman): (a) objective function and (b) MSE.

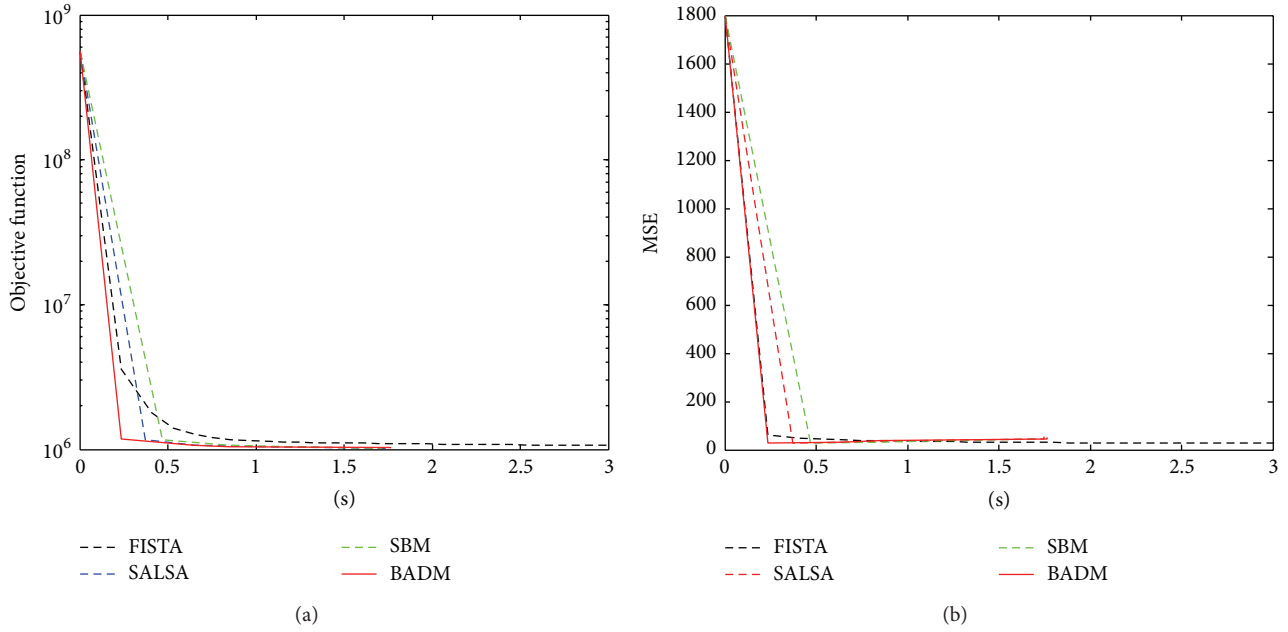


FIGURE 6: Evolutions in experiment (II) (Cameraman): (a) objective function and (b) MSE.

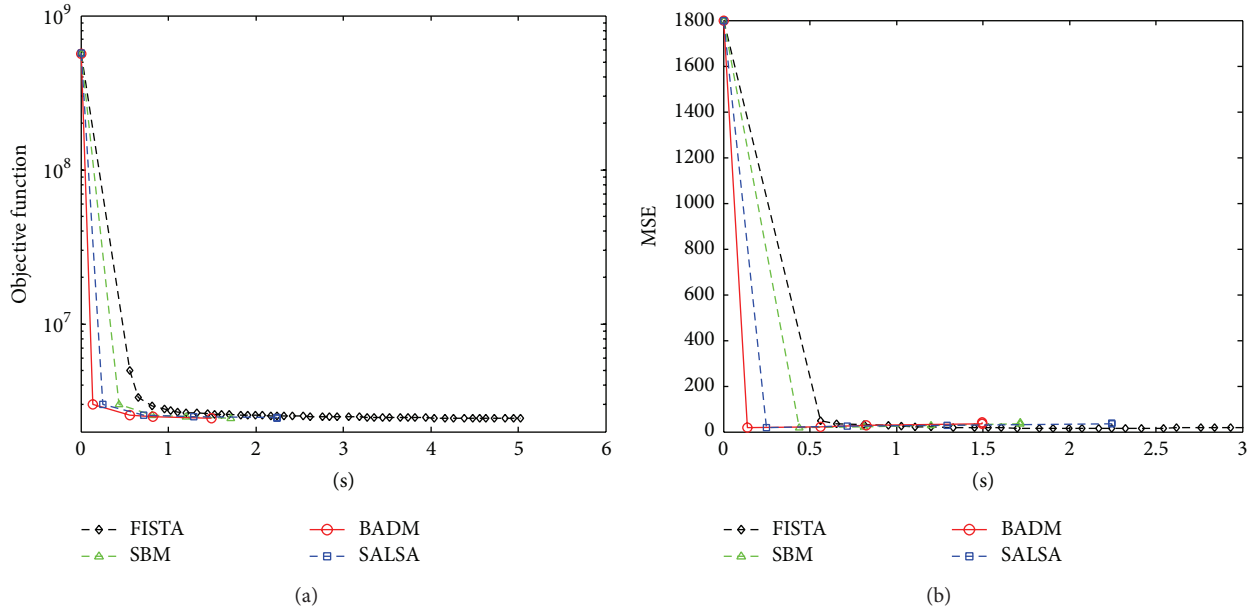


FIGURE 7: Evolutions in experiment (III) (Cameraman): (a) objective function and (b) MSE.

yielding the following *bound alternative direction method* (BADM) (Algorithm 4).

In (Algorithm 4), Step 3 is equivalent to $\text{Prox}_{f/\beta}(\mathbf{x}^k + \mathbf{d}^k)$, while Step 4 (see (13) and (9)) is equivalent to

$$\text{Prox}_{\Phi_{\epsilon^k}(\mathbf{x},\mathbf{x}^k)/\beta}\left(\mathbf{v}^{k+1} - \mathbf{d}^k\right) = \text{soft}\left(\mathbf{v}^{k+1} - \mathbf{d}^k, \frac{\boldsymbol{\omega}_{\epsilon^k}\left(\mathbf{x}^k\right)}{\beta}\right). \tag{23}$$

Moreover, since the objective function of (10) is nonconvex for $p \in (0, 1)$, BADM cannot be guaranteed to converge to a global optimum. Nevertheless, as stated in Theorem 3, the proposed algorithm is able to obtain a stationary point, which in practice is always a good quality deblurred image.

3. Image Deblurring Using Synthesis ℓ_p Formulation and BADM

In this section, the synthesis ℓ_p formulation (see (4)) and the BADM are applied to image deblurring. Note that using the analysis ℓ_p formulation can be naturally extended

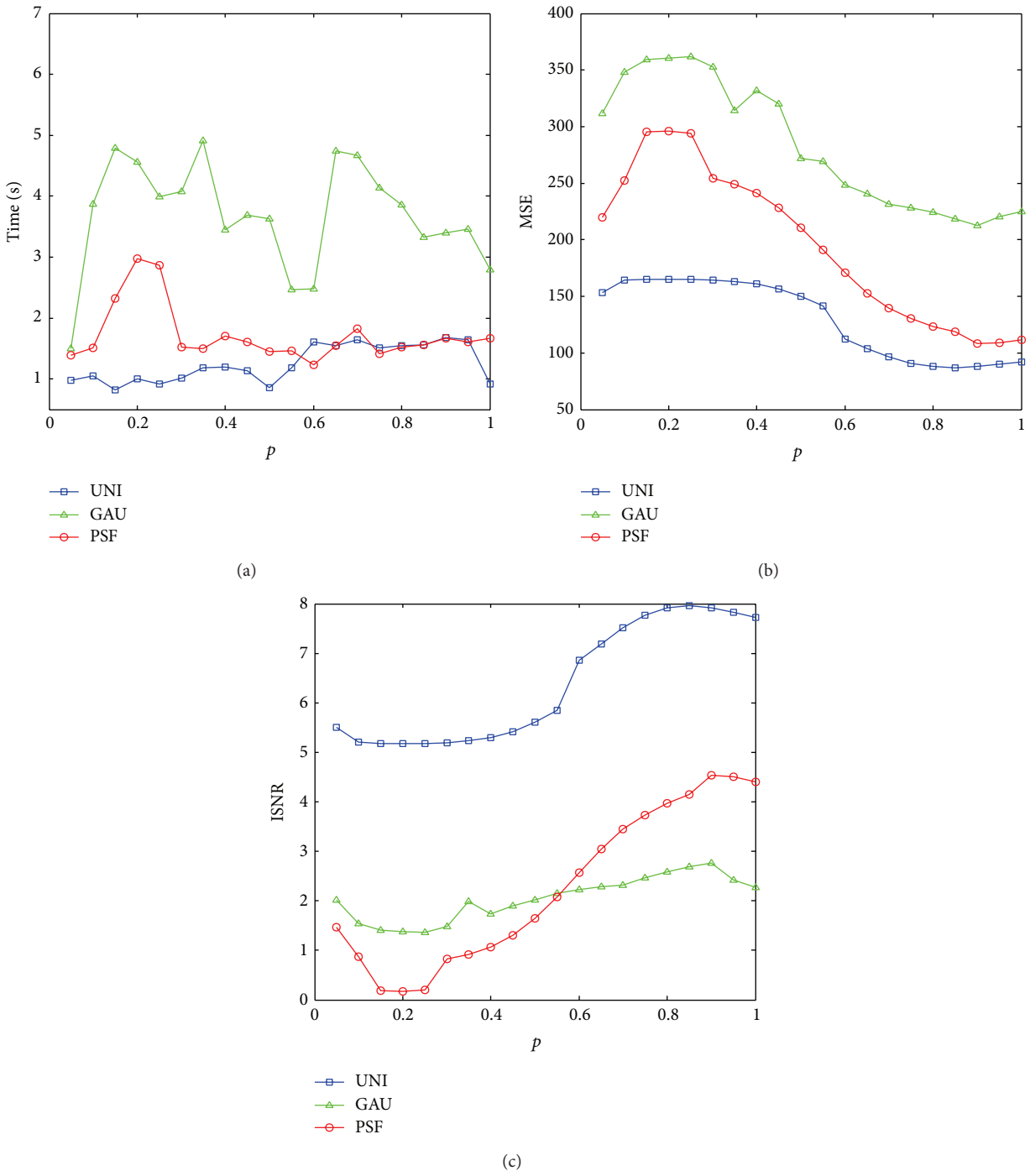


Figure 8: Performance of BADM over p on deblurring the corrupted Cameraman images: (a) time; (b) MSE; and (c) ISNR.

Table 1: Results of deblurring the Cameraman images.

Algorithm	Time (seconds)			Iterations			MSE			ISNR		
	UNI	GAU	PSF	UNI	GAU	PSF	UNI	GAU	PSF	UNI	GAU	PSF
FISTA	16.52	11.01	5.02	150	98	47	102.71	217.85	112.32	7.19	2.67	4.39
SALSA	**0.80**	**2.17**	1.72	**4**	**8**	**6**	90.40	214.41	109.25	7.80	2.75	4.52
SBM	0.95	2.53	**1.50**	**4**	**8**	**6**	90.40	214.36	109.25	7.80	2.75	4.52
BADM	1.44	3.28	1.55	**4**	9	**6**	**85.42**	**212.29**	**105.18**	**8.05**	**2.78**	**4.66**

Table 2: Results of deblurring the Mandril images.

Algorithm	Time (seconds)			Iterations			MSE			ISNR		
	UNI	GAU	PSF	UNI	GAU	PSF	UNI	GAU	PSF	UNI	GAU	PSF
FISTA	56.82	49.11	17.22	110	94	33	119.23	288.45	142.12	5.50	1.42	2.19
SALSA	3.41	5.41	3.63	**4**	**8**	**5**	119.12	255.30	82.98	5.59	1.95	4.51
SBM	**3.06**	**4.51**	**2.81**	**4**	**8**	**5**	119.63	255.30	82.98	5.59	1.95	4.51
BADM	4.26	9.34	3.60	**4**	**8**	**5**	**117.24**	**246.12**	**81.15**	**5.69**	**2.11**	**4.60**

Table 3: Results of deblurring the Lena images.

Algorithm	Time (seconds)			Iterations			MSE			ISNR		
	UNI	GAU	PSF	UNI	GAU	PSF	UNI	GAU	PSF	UNI	GAU	PSF
FISTA	53.13	40.8	21.41	98	81	42	43.89	82.84	53.90	6.01	2.82	2.17
SALSA	2.91	4.95	4.09	**4**	6	**6**	41.15	72.42	34.32	6.15	3.34	4.14
SBM	**2.52**	**4.07**	**3.42**	**4**	6	**6**	41.15	72.42	34.32	6.15	3.34	4.14
BADM	4.22	5.21	6.66	**4**	**5**	**6**	**38.06**	**69.10**	**34.10**	**6.47**	**3.59**	**4.18**

and will be left as future work. Therefore, here $f(\mathbf{s}) = (1/2)\|\mathbf{AWs} - \mathbf{y}\|_2^2$, and $\Phi_{\epsilon^k}(\mathbf{s}, \mathbf{s}^k)$ is the proposed majorizer of $\lambda\|\mathbf{s}\|_p^p$ at k iteration, which are inserted into the BADM yielding (Algorithm 5), where Step 4 is derived from (18). Assume that $\mathbf{A}$ represents a (periodic) convolution and $\mathbf{W}$ is a normalized Parseval frame (i.e., $\mathbf{WW}^T = \mathbf{I}$, where $\mathbf{I}$ is the identity matrix and possibly $\mathbf{W}^T\mathbf{W} \neq \mathbf{I}$). According to the Sherman-Morrison-Woodbury matrix inversion formula,

$$\left[(\mathbf{AW})^T\mathbf{AW} + \beta\mathbf{I}\right]^{-1} = \frac{1}{\beta}\left[\mathbf{I} - \mathbf{W}^T\mathbf{A}^T\left(\mathbf{AA}^T + \beta\mathbf{I}\right)^{-1}\mathbf{AW}\right]. \tag{24}$$

Moreover, under the periodic boundary condition for $\mathbf{s}$, $\mathbf{AA}^T$ is block circulant, such that $(\mathbf{AA}^T + \beta\mathbf{I})^{-1}$ can be diagonalized by two-dimensional discrete Fourier transform (DFT). Let $\mathbf{F} = \mathbf{A}^T(\mathbf{AA}^T + \beta\mathbf{I})^{-1}\mathbf{A}$; then $\mathbf{F}$ is equivalent to a filter in the Fourier domain and the cost of products by $\mathbf{F}$ using FFT algorithms is $\mathcal{O}(n\log n)$ [10]. Thus,

$$\mathbf{v}^{k+1} = \frac{1}{\beta}\left(\mathbf{I} - \mathbf{W}^T\mathbf{FW}\right)\mathbf{u}^k, \tag{25}$$

where $\mathbf{u}^k = (\mathbf{AW})^T\mathbf{y} + \beta(\mathbf{s}^k + \mathbf{d}^k)$. As the computational complexity analysis in [10], the cost of computing $\mathbf{v}^{k+1}$ is $\mathcal{O}(n\log n)$. Moreover, computing $\mathbf{s}^{k+1}$ is the soft thresholding whose cost is $\mathcal{O}(n)$ and computing $\mathbf{d}^{k+1}$ also has $\mathcal{O}(n)$ cost. Therefore, each iteration of the BADM for (4) has $\mathcal{O}(n\log n)$ cost.

4. Experiments

In this section, the proposed algorithm BADM for (4) is compared with the state-of-the art algorithms: FISTA [14], SALSA [11], and SBM [22] for (2) (from now on, the BADM is only for (4) while FISTA, SALSA, and SBM are for (2) if without specification). Consider the low-frequency images (Cameraman), high-frequency images (Mandril), and both low- and high-frequency images (Lena) (see Figure 1), with size 512 × 512 pixels, corrupted by the following three benchmark cases [5, 11]: (I) uniform blur of size 9 × 9 and noise variance $\sigma^2 = 0.56^2$ (termed UNI); (II) Gaussian blur kernel with $\sigma^2 = 8$ (termed GAU); (III) the point spread function of the blur operator is $h_{ij} = 1/(1 + i^2 + j^2)$ for $i, j = -7, \ldots, 7$ and $\sigma^2 = 8$ (termed PSF). All the experiments were performed using MATLAB on a 64-bit Windows 7 PC with an Intel Core i7 3.07 GHz processor and 6.0 GB of RAM. In order to measure the performance of different algorithms, the following four metrics (where $\mathbf{x}$ is the original image and $\mathbf{y}^k$ and $\mathbf{x}^k$ are the observed image and the estimated image at the k iteration, resp.) are employed: (a) consumed CPU time (Time); (b) number of iterations (Iterations); (c) mean square error (MSE $= \|\mathbf{x} - \mathbf{x}^k\|^2/n$); and (d) improvement in SNR (ISNR $= 10\log_{10}(\sum_k \|\mathbf{x} - \mathbf{y}^k\|^2 / \sum_k \|\mathbf{x} - \mathbf{x}^k\|^2)$) and the stopping criterion is chosen as $\|\mathbf{x}^{k+1} - \mathbf{x}^k\|/\|\mathbf{x}^{k+1}\| \leq \eta$, where $p = 0.9$, and η, as well as other necessary parameters (such as λ, β for the proposed BADM algorithm), is hand tuned for each algorithm in each experiment for the best ISNR; that is, $\lambda = 0.0075$ for experiment (I); 0.09 for (II); and 0.25 for (III) and $\beta = \lambda/10$ in (I), (II), and (III).

The obtained results of four metrics for the Cameraman, Mandril, and Lena images are listed in Tables 1, 2, and 3, respectively, and the deblurred images by BADM are shown in Figures 2, 3, and 4, respectively. And the evolutions of objective function and MSE by different algorithms in experiments (I), (II), and (III) (to avoid redundancy, only for the results of Cameraman images) are shown in Figures 5, 6, and 7, respectively. Moreover, the results of time, MSE, and ISNR over p are shown in Figure 8. From above results, it is clear that the BADM outperforms the FISTA, SALSA, and SBM in terms of MSE and ISNR.

5. Conclusions

This paper has proposed a new *bound alternative direction method* (BADM) for the ℓ_p ($p \in (0, 1)$) minimization

problems in image deblurring. In order to solve the unconstrained ℓ_p optimization problem, the idea of BADM is to first bound the ℓ_p regularizer to obtain a bound unconstrained problem, which is then reformulated into a constrained one by variable splitting. The resulting constrained problem is further addressed by an augmented Lagrangian method, more specifically, the *alternating direction method of multipliers* (ADMM). Therefore, the BADM is an extension of the ADMM to solve the ℓ_p ($p \in (0, 1)$) minimization problems. Experiments on a set of image deblurring problems have shown that the BADM for the synthesis ℓ_p formulation is favorably competitive with the state-of-the-art algorithms for the synthesis ℓ_1 formulation.

In future work, the BADM will be applied to the analysis ℓ_p formulation and other applications such as in painting and magnetic resonance imaging.

Conflict of Interests

The author declares that there is no conflict of interests regarding to the publication of this paper.

Acknowledgments

This work was partially supported by the China Scholarship Council (CSC: 2010611017). Xiangrong Zeng would like to thank the anonymous reviewers who have helped to improve the quality of this paper.

References

[1] C. Chaux, P. L. Combettes, J.-C. Pesquet, and V. R. Wajs, "A variational formulation for frame-based inverse problems," *Inverse Problems*, vol. 23, article 1495, no. 4, 2007.

[2] I. Daubechies, M. Defrise, and C. de Mol, "An iterative thresholding algorithm for linear inverse problems with a sparsity constraint," *Communications on Pure and Applied Mathematics*, vol. 57, no. 11, pp. 1413–1457, 2004.

[3] M. Elad, B. Matalon, and M. Zibulevsky, "Image denoising with shrinkage and redundant representations," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, vol. 2, pp. 1924–1931, June 2006.

[4] M. A. T. Figueiredo and R. D. Nowak, "A bound optimization approach to wavelet-based image deconvolution," in *Proceedings of the IEEE International Conference on Image Processing (ICIP '05)*, vol. 2, pp. 782–785, September 2005.

[5] M. A. T. Figueiredo and R. D. Nowak, "An EM algorithm for wavelet-based image restoration," *IEEE Transactions on Image Processing*, vol. 12, no. 8, pp. 906–916, 2003.

[6] J. M. Bioucas-Dias, "Bayesian wavelet-based image deconvolution: a gem algorithm exploiting a class of heavy-tailed priors," *IEEE Transactions on Image Processing*, vol. 15, no. 4, pp. 937–951, 2006.

[7] M. A. T. Figueiredo, J. M. Bioucas-Dias, and R. D. Nowak, "Majorization-minimization algorithms for wavelet-based image restoration," *IEEE Transactions on Image Processing*, vol. 16, no. 12, pp. 2980–2991, 2007.

[8] T. Chan, S. Esedoglu, F. Park, and A. Yip, "Recent developments in total variation image restoration," in *Mathematical Models of Computer Vision*, vol. 17, 2005.

[9] J.-F. Cai, B. Dong, S. Osher, and Z. Shen, "Image restoration: total variation, wavelet frames, and beyond," *Journal of the American Mathematical Society*, vol. 25, no. 4, pp. 1033–1089, 2012.

[10] M. V. Afonso, J. M. Bioucas-Dias, and M. A. T. Figueiredo, "An augmented Lagrangian approach to the constrained optimization formulation of imaging inverse problems," *IEEE Transactions on Image Processing*, vol. 20, no. 3, pp. 681–695, 2011.

[11] M. V. Afonso, J. M. Bioucas-Dias, and M. A. T. Figueiredo, "Fast image recovery using variable splitting and constrained optimization," *IEEE Transactions on Image Processing*, vol. 19, no. 9, pp. 2345–2356, 2010.

[12] M. Elad, P. Milanfar, and R. Rubinstein, "Analysis versus synthesis in signal priors," *Inverse Problems*, vol. 23, no. 3, pp. 947–968, 2007.

[13] J. M. Bioucas-Dias and M. A. T. Figueiredo, "A new TwIST: two-step iterative shrinkage/thresholding algorithms for image restoration," *IEEE Transactions on Image Processing*, vol. 16, no. 12, pp. 2992–3004, 2007.

[14] A. Beck and M. Teboulle, "A fast iterative shrinkage-thresholding algorithm for linear inverse problems," *SIAM Journal on Imaging Sciences*, vol. 2, no. 1, pp. 183–202, 2009.

[15] S. J. Wright, R. D. Nowak, and M. A. T. Figueiredo, "Sparse reconstruction by separable approximation," *IEEE Transactions on Signal Processing*, vol. 57, no. 7, pp. 2479–2493, 2009.

[16] Y. Nesterov, *Introductory Lectures on Convex Optimization*, 2004.

[17] Y. Nesterov, "A method of solving a convex programming problem with convergence rate o (1/k2)," *Soviet Mathematics Doklady*, vol. 27, no. 2, pp. 372–376, 1983.

[18] J. Eckstein and D. P. Bertsekas, "On the Douglas-Rachford splitting method and the proximal point algorithm for maximal monotone operators," *Mathematical Programming*, vol. 55, no. 1–3, pp. 293–318, 1992.

[19] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein, "Distributed optimization and statistical learning via the alternating direction method of multipliers," *Foundations and Trends in Machine Learning*, vol. 3, no. 1, pp. 1–122, 2010.

[20] S. Setzer, "Split bregman algorithm, douglas-rachford splitting and frame shrinkage," in *Scale Space and Variational Methods in Computer Vision*, vol. 5567 of *Lecture Notes in Computer Science*, pp. 464–476, Springer, Berlin, Germany, 2009.

[21] W. Yin, S. Osher, D. Goldfarb, and J. Darbon, "Bregman iterative algorithms for ℓ_1-minimization with applications to compressed sensing," *SIAM Journal on Imaging Sciences*, vol. 1, no. 1, pp. 143–168, 2008.

[22] T. Goldstein and S. Osher, "The split Bregman method for $L1$-regularized problems," *SIAM Journal on Imaging Sciences*, vol. 2, no. 2, pp. 323–343, 2009.

[23] S. Osher, Y. Mao, B. Dong, and W. Yin, "Fast linearized Bregman iteration for compressive sensing and sparse denoising," *Communications in Mathematical Sciences*, vol. 8, no. 1, pp. 93–111, 2010.

[24] A. Langer and M. Fornasier, "Analysis of the adaptive iterative bregman algorithm," preprint, 2010.

[25] H. Zhang, L. Cheng, and J. Li, "Reweighted minimization model for MR image reconstruction with split Bregman method," *Science China: Information Sciences*, vol. 55, no. 9, pp. 2109–2118, 2012.

[26] J. Cai, S. Osher, and Z. Shen, "Split bregman methods and frame based image restoration," *Multiscale Modeling and Simulation*, vol. 8, no. 2, pp. 337–369, 2009.

[27] S. Setzer, G. Steidl, and T. Teuber, "Deblurring Poissonian images by split Bregman techniques," *Journal of Visual Communication and Image Representation*, vol. 21, no. 3, pp. 193–199, 2010.

[28] E. Candes, M. Wakin, and S. Boyd, "Enhancing sparsity by reweighted l_1 minimization," *Journal of Fourier Analysis and Applications*, vol. 14, no. 5, pp. 877–905, 2008.

[29] R. Chartrand, "Exact reconstruction of sparse signals via nonconvex minimization," *IEEE Signal Processing Letters*, vol. 14, no. 10, pp. 707–710, 2007.

[30] R. Chartrand and W. Yin, "Iteratively reweighted algorithms for compressive sensing," in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP '08)*, pp. 3869–3872, Las Vegas, Nev, USA, April 2008.

[31] R. Saab, R. Chartrand, and Ö. Yilmaz, "Stable sparse approximations via nonconvex optimization," in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP '08)*, pp. 3885–3888, April 2008.

[32] Z. Xu, X. Chang, F. Xu, and H. Zhang, "$L_{1/2}$ regularization: a thresholding representation theory and a fast solver," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 23, no. 7, pp. 1013–1027, 2012.

[33] R. Chartrand and V. Staneva, "Restricted isometry properties and nonconvex compressive sensing," *Inverse Problems*, vol. 24, no. 3, Article ID 035020, 14 pages, 2008.

[34] S. Foucart and M. Lai, "Sparsest solutions of underdetermined linear systems via l_q-minimization for $0 < q < 1$," *Applied and Computational Harmonic Analysis*, vol. 26, no. 3, pp. 395–407, 2009.

[35] Q. Sun, "Recovery of sparsest signals via ℓq-minimization," *Applied and Computational Harmonic Analysis*, vol. 32, no. 3, pp. 329–341, 2012.

[36] X. Chen and W. Zhou, "Convergence of reweighted $\ell 1$ minimization algorithms and unique solution of truncated ℓp minimization," Tech. Rep., The Hong Kong Polytechnic University, 2010.

[37] D. Needell, "Noisy signal recovery via iterative reweighted L_1-minimization," in *Proceedings of the 43rd Asilomar Conference on Signals, Systems and Computers*, pp. 113–117, Pacific Grove, Calif, USA, November 2009.

[38] I. Daubechies, R. DeVore, M. Fornasier, and C. S. Güntürk, "Iteratively reweighted least squares minimization for sparse recovery," *Communications on Pure and Applied Mathematics*, vol. 63, no. 1, pp. 1–38, 2010.

[39] D. Wipf and S. Nagarajan, "Iterative reweighted ℓ_1 and ℓ_2 methods for finding sparse solutions," *IEEE Journal on Selected Topics in Signal Processing*, vol. 4, no. 2, pp. 317–329, 2010.

[40] M. A. Khajehnejad, W. Xu, A. S. Avestimehr, and B. Hassibi, "Improved sparse recovery thresholds with two-step reweighted $\ell 1$ minimization," in *Proceeding of the IEEE International Symposium on Information Theory (ISIT '10)*, pp. 1603–1607, Austin, Tex, USA, June 2010.

[41] R. Courant, "Variational methods for the solution of problems of equilibrium and vibrations," *Bulletin of the American Mathematical Society*, vol. 49, no. 1, pp. 1–23, 1943.

[42] Y. Wang, J. Yang, W. Yin, and Y. Zhang, "A new alternating minimization algorithm for total variation image reconstruction," *SIAM Journal on Imaging Sciences*, vol. 1, no. 3, pp. 248–272, 2008.

[43] J. Nocedal and S. Wright, *Numerical Optimization*, Springer, 1999.

[44] H. H. Bauschke and P. L. Combettes, *Convex Analysis and Monotone Operator Theory in Hilbert Spaces*, CMS Books in Mathematics, Springer, New York, NY, USA, 2011.

[45] Z. Lu, "Iterative reweighted minimization methods for l_p regularized unconstrained nonlinear programming," *Mathematical Programming*, pp. 1–31, 2012.

[46] L. Grippo and M. Sciandrone, "On the convergence of the block nonlinear Gauss-Seidel method under convex constraints," *Operations Research Letters*, vol. 26, no. 3, pp. 127–136, 2000.

Hindawi Publishing Corporation
Mathematical Problems in Engineering
Volume 2014, Article ID 545689, 9 pages
http://dx.doi.org/10.1155/2014/545689

Research Article

Resolving Power of Algorithm for Solving the Coefficient Inverse Problem for the Geoelectric Equation

K. T. Iskakov and Zh. O. Oralbekova

L.N. Gumilyov Eurasian National University, Faculty of Information Technologies, Astana 010008, Kazakhstan

Correspondence should be addressed to Zh. O. Oralbekova; oralbekova@bk.ru

Received 9 April 2014; Revised 27 June 2014; Accepted 13 July 2014; Published 6 August 2014

Academic Editor: Valery G. Yakhno

We considered the inverse coefficient problem for the geoelectric equation. For the purpose of research of the conditional stability of the inverse problem solution, we used integral formulation of the inverse geoelectric problem. By implementing the relevant norms and using the close system of Volterra integral equations, we managed to estimate the conditional stability of the solution of inverse problem or rather lower changes in input data imply lower changes in the solution (of the numerical method). When determining the additional information the device errors are possible. That is why this research is important for experimental studies with usage of ground penetrating radars.

1. Introduction

Inverse problems for hyperbolic equations, in particular for the acoustics and geoelectrics, were investigated by many authors; notably, a detailed bibliography is given in the monography of Kabanikhin [1]. We will present the main scientific results on this problem. Blagoveshchenskii applied Gelfand-Levitan method for proving the uniqueness of the solution of the inverse acoustic problem [2]. Romanov proved a comparable theorem for the following equation [3]:

$$w_{tt}(x,t) = w_{xx}(x,t) - q(x)\,w(x,t), \tag{1}$$

which is consolidated from the acoustic equation with well-known transformation (see [4]):

$$\begin{aligned} w(x,t) &= u(x,t)\exp\left\{-\frac{1}{2}\ln\sigma(x)\right\}, \\ q(x) &= -\frac{1}{2}[\ln\sigma(x)]'' + \frac{1}{4}\left[\frac{\sigma'(x)}{\sigma(x)}\right]^2. \end{aligned} \tag{2}$$

Romanov and Yamamoto [5] obtained the estimation of conditional stability in L_2 for getting a multidimension analog of the inverse problem (1).

Numerical algorithm of inverse acoustic problem solving in the discrete case was given in work [6] for the first time.

Bamberger and his coauthors used a conjugate gradient method to define the acoustic impedance [7, 8].

He and Kabanikhin used the optimization method to solve the inverse problem for three-dimension acoustic equation [9].

Azamatov and Kabanikhin studied the conditional stability of the solution to Volterra operator equation in L_2 [10].

Problems of uniqueness of the inverse problem solution and set of numerical methods for solving the geoelectric equation were given in the monograph of Romanov and Kabanikhin [11].

For solving inverse acoustic problem in integral case formulation the estimation of the conditional stability in H^1 was obtained in the work of Kabanikhin et al. [12].

Further, in works [13, 14] for minimizing purposes they built and investigated a special form of the composite functional that allowed proving the following theorems in the space L_2: the local correctness theorem, the correctness theorem of the inverse problem for small amount of data, and the correctness theorem in the envelope of the exact solution in L_2.

Bukhgeim and Klibanov suggested using the method of Carleman estimates when proving uniqueness theorems of the coefficient inverse problems [15]. A broad overview on the use of Carleman estimates in the theory of multidimension coefficient inverse problems is given in the work [16].

The problem of uniqueness of inverse problem solution for determination of the coefficients of the permittivity and conductivity for Maxwell's equation system is considered in the work [17].

Approbation of the globally convergent numerical algorithm with the use of experimental radar data for determination of the permittivity is given in work [18]. They presented an analysis of convergence of the method and it has been shown that the computed and real values of permittivity were in enough agreement. A wide range of globally convergent algorithms of solving a class of problems is described in work [19].

Comparative analysis of the classical equation methods and globally convergent numerical method of solving the coefficient inverse problems was given in work [20]. These comparisons were performed for both computationally simulated and experimental data.

In the work [21] continuation problem from the time-like surface for the 2D Maxwell's equation was considered. The gradient method for the continuation and coefficient inverse problem was explained. The results of computational experiment were presented.

In this research, following the methods which were described in the work [12], we obtained the estimation results of the conditional stability of the geoelectric equation in H^1.

Herein after the second paragraph there is the conclusion of the main equations which were derived from the system of Maxwell's equations [11].

In the third paragraph we had amplified the inverse problem for the geoelectric equation with data on characteristics. It allows us to obtain a close system of integral equations.

Finally, in the fourth paragraph, the implementation of the relevant class of input data functions and the class of solutions of the inverse problem allowed us to estimate the conditional stability of the inverse problem solution for the geoelectric equation.

2. Statement of the Problems

The propagation process of electromagnetic waves in a medium is described by Maxwell's equations [11]:

$$\begin{aligned} &\varepsilon\frac{\partial}{\partial t}E - \operatorname{rot} H + \sigma E + j^{\mathrm{cm}} = 0,\\ &x_3 \neq 0, \quad (x_1, x_2, x_3) \in R^3,\\ &\mu\frac{\partial}{\partial t}H + \operatorname{rot} E = 0, \quad t > 0. \end{aligned} \tag{3}$$

Here $E = (E_1, E_2, E_3)^*$ and $H = (H_1, H_2, H_3)^*$ are the electric and magnetic fields intensity vectors; ε is dielectric permittivity of the medium; μ is magnetic permeability of the medium; σ is conductivity of the medium; j^{cm} is source of external currents.

Consider geophysical model of the medium consisting of two half spaces: $R^3_- = \{x \in R^3, x_3 < 0\}$—air; $R^3_+ = \{x \in R^3, x_3 > 0\}$—earth.

Let the external current source take the following form:

$$j^{\mathrm{cm}} = (0, 1, 0)^* g(x_1)\,\delta(x_3)\,\theta(t), \tag{4}$$

where $g(x_1)$ is the function which describes the transversal dimension of the source; $\delta(x_3)$ is Dirac delta function; and $\theta(t)$ is Heaviside function.

Setting the external current in the form (4) makes it an instantaneous inclusion current, parallel to the axis x_2 at time scales of 10–50 ns (nanoseconds).

Using the definition of the curl we get finally from Maxwell's equations

$$\begin{aligned} \varepsilon\frac{\partial}{\partial t}E_1 + \sigma E_1 &= \frac{\partial}{\partial x_2}H_3 - \frac{\partial}{\partial x_3}H_2,\\ \varepsilon\frac{\partial}{\partial t}E_2 + \sigma E_2 &= -\frac{\partial}{\partial x_1}H_3 + \frac{\partial}{\partial x_3}H_1 + \gamma_2,\\ \varepsilon\frac{\partial}{\partial t}E_3 + \sigma E_3 &= \frac{\partial}{\partial x_1}H_2 - \frac{\partial}{\partial x_2}H_1,\\ \mu\frac{\partial}{\partial t}H_1 &= -\frac{\partial}{\partial x_2}E_3 + \frac{\partial}{\partial x_3}E_2,\\ \mu\frac{\partial}{\partial t}H_2 &= -\frac{\partial}{\partial x_1}E_3 - \frac{\partial}{\partial x_3}E_1,\\ \mu\frac{\partial}{\partial t}H_3 &= -\frac{\partial}{\partial x_1}E_2 - \frac{\partial}{\partial x_2}E_1. \end{aligned} \tag{5}$$

Assuming that the coefficients of Maxwell's equations do not depend on the variable x_2 and are of the special choice of the source in the form (4), the system will retain only three nonzero components E_2, H_1, and H_3 [11]. Excluding the last two components, the final equations are written such that

$$\begin{aligned} &\varepsilon\frac{\partial^2}{\partial t^2}E_2 + \sigma\frac{\partial}{\partial t}E_2\\ &\quad = \frac{\partial}{\partial x_1}\left(\frac{1}{\mu}\frac{\partial}{\partial x_1}E_2\right) + \frac{\partial}{\partial x_3}\left(\frac{1}{\mu}\frac{\partial}{\partial x_3}E_2\right)\\ &\qquad + g(x_1)\,\eta(x_3)\,\theta'(t), \quad x_3 > 0,\ t > 0, \end{aligned} \tag{6}$$

$$E_2\big|_{t>0} = 0, \tag{7}$$

$$E_2\big|_{x_3=+0} = \varphi_{(1)}(x_1, t), \tag{8}$$

$$\left(\frac{1}{\mu}\frac{\partial}{\partial x_3}E_2\right)\Big|_{x_3=+0} = \frac{\partial}{\partial t}\varphi_{(2)}(x_3, t). \tag{9}$$

Particular attention has aggravated conditions (8) and (9).

Condition (8) is taken as additional information (the response of the medium).

Condition (9) is unknown, but it is necessary for solving direct and inverse problems in a half space $\{x_3 > 0\}$ (earth).

In this situation we proceed as shown in [11], in the half space $\{x_3 \le 0\}$ where $\sigma = 0$ we solve the direct problem by the known data ε, μ:

$$\varepsilon \frac{\partial^2}{\partial t^2} E_2 = \frac{\partial}{\partial x_1}\left(\frac{1}{\mu}\frac{\partial}{\partial x_1}E_2\right) + \frac{\partial}{\partial x_3}\left(\frac{1}{\mu}\frac{\partial}{\partial x_3}E_2\right) + g(x_1)\eta(x_3)\theta'(t), \quad x_3 < 0,\ t > 0, \tag{10}$$

$$E_2|_{t<0} = 0, \tag{11}$$

$$E_2|_{x_3=+0} = \varphi_{(1)}(x_1, t). \tag{12}$$

In the last system we consider known additional information (8) as a boundary condition for solving the direct problem in the area $\{x_3 < 0\}$ (air). This fact enables us to restrict the numerical solution of the inverse problem for the minimum possible size of the area in the plane $\{x_3 > 0\}$.

If the coefficients of (10) do not depend on the variable x_1 [11] then applying the Fourier transform $F_{x_1}[\cdot]$ to (10)–(12) and similar to (6)–(9), we write the final statement of the problem.

In the air domain $\{x_3 < 0\}$ we have the following statement of the direct problem:

$$\varepsilon \tilde{v}_{tt} = \frac{1}{\mu}\tilde{v}_{x_3x_3} - \frac{\lambda^2}{\mu}\tilde{v} + \tilde{g}_\lambda \eta(x_3)\theta'(t), \quad x_3 < 0, \qquad \tilde{v}|_{t<0} = 0, \qquad \tilde{v}_t|_{t<0} = 0, \qquad \tilde{v}(0,t) = f_{(1)}(t). \tag{13}$$

In the earth domain $\{x_3 > 0\}$ we have the following statement of the direct problem:

$$\tilde{v}_{tt} + \frac{\sigma}{\varepsilon}\tilde{v}_t = \frac{1}{\mu\varepsilon}\tilde{v}_{x_3x_3} - \frac{\lambda^2}{\mu\varepsilon}\tilde{g}_\lambda \delta(x_3, t), \quad x_3 > 0,\ x_3 \in R^1, \tag{14}$$

$$\tilde{v}|_{t<0} = 0, \qquad \tilde{v}_t|_{t<0} = 0, \tag{15}$$

$$\frac{1}{\mu}\tilde{v}_{x_3}(0,t) = f_{(2)}(t), \tag{16}$$

$$\tilde{v}(0,t) = f_{(1)}(t). \tag{17}$$

Here λ is a Fourier parameter and $\tilde{v}(x,t) = F_{x_1}[E_2(x_1, 0, x_3, t)]$; $f_{(1)}(t) = F_{x_1}[\varphi_{(1)}(x_1,t)]$; and $f_{(2)}(t) = F_{x_1}[(\partial/\partial t)\varphi_{(2)}(x_1,t)]$ are Fourier images.

Direct Problem. By the known values of ε, μ, and σ find $\tilde{v}(x_3,t)$ as the solution of the mixed problem (14)–(16).

Inverse Problem. Find $\sigma(x_3)$ and $\tilde{v}(x_3,t)$ from (14) to (16) for given $f_{(1)}(t)$ with fixed $\lambda = \lambda_0$.

To study conditional stability of the inverse geoelectric problem, it is convenient to use the integral formulation.

Now we introduce the following notations: $b(x_3) = 1/\mu\varepsilon(x_3)$ and $a(x_3) = \sigma(x_3)/\varepsilon(x_3)$ and change the variables and functions:

$$z = z(x_3) = \int_0^{x_3} \sqrt{\mu\varepsilon(\xi)}\,d\xi, \quad x_3 = \omega(z); \qquad a(z) = \frac{\sigma(\omega(z))}{\varepsilon(\omega(z))}, \qquad b(z) = \frac{1}{\mu\varepsilon(\omega(z))}, \qquad u(z,t) = v(\omega(z),t), \qquad j_0 = -g(\lambda)\sqrt{\frac{\mu}{\varepsilon(0)}}. \tag{18}$$

Then (14)–(16) can be written in the form

$$u_{tt}(z,t) = u_{zz}(z,t) - a(z)u_t(z,t) - \frac{b'(z)}{b(z)}u_z(z,t) - (\lambda b(z))^2 u(z,t), \tag{19}$$

$$u|_{t<0} = 0, \qquad u_t|_{t<0} = 0, \tag{20}$$

$$\frac{1}{\mu}u_z(0,t) = f_{(2)}(t), \tag{21}$$

$$u(0,t) = f_{(1)}(t). \tag{22}$$

In the future, we will get (19) without the derivative u_z; for this we assume that

$$u(z,t) = G(z)\,v(z,t). \tag{23}$$

Now we calculate derivatives as follows:

$$u_z = G'v + Gv_z, \qquad u_{zz} = G''v + 2G'v_z + Gv_{zz}, \qquad u_t = Gv_t, \qquad u_{tt} = Gv_{tt}. \tag{24}$$

Substituting (24) into (19), we obtain

$$Gv_{tt} = G''v + 2G'v_z + Gv_{zz} - a(z)Gv_t - \frac{b'}{b}\left(G'v + Gv_z\right) - (\lambda b)^2 Gv. \tag{25}$$

Grouped together, we obtain

$$v_{tt} = v_{zz} + \left(2\frac{G'}{G} - \frac{b'}{b}\right)v_z - a(z)v_t + \left(\frac{G''}{G} - \frac{b'}{b}\frac{G'}{G} - (\lambda b)^2\right)v. \tag{26}$$

Put that

$$2\frac{G'}{G} - \frac{b'}{b} = 0, \tag{27}$$

$$g(z) = \frac{G''}{G} - \frac{b'}{b}\frac{G'}{G} - (\lambda b)^2. \tag{28}$$

Finally, we have

$$\begin{aligned} v_{tt} &= v_{zz} - a(z)\, v_t + g(z)\, v, \\ v|_{t<0} &= 0, \qquad v_t|_{t<0} = 0, \\ \frac{1}{\mu} v_z(0,t) &= f_{(2)}(t), \\ v(0,t) &= f_{(1)}(t). \end{aligned} \tag{29}$$

From (27) we have

$$\begin{aligned} &\frac{G'}{G} = \frac{b'}{2b}; \qquad (\ln G)' = \left(\ln \sqrt{b}\right)', \\ &\ln G = \ln \sqrt{b} + \ln S(0), \quad S(0) = 1, \\ &G(z) = \sqrt{b(z)}. \end{aligned} \tag{30}$$

Thus, the function $g(z)$ is uniquely determined from (28) by the formula (30).

3. Statement of the Problem with the Data on the Characteristics

In the domain $\Delta(l) = \{(z,t) | 0 < |z| < t < l\}$ we consider the inverse problem with data on the characteristics [11]:

$$v_{tt}(z,t) = v_{zz}(z,t) - Pv(z,t), \quad (z,t) \in \Delta l, \tag{31}$$

$$v(z,z) = S(z), \quad 0 \le z \le l, \tag{32}$$

$$v(0,t) = f(t), \quad 0 \le t \le 2l, \tag{33}$$

$$v_z(0,t) = \varphi(t), \quad 0 \le t \le 2l. \tag{34}$$

Here

$$\begin{aligned} Pv(z,t) &= a(z)\, v_t(z,t) + g(z)\, v(z,t), \\ f(t) &= f_{(1)}(t), \quad \varphi(t) = \mu f_{(2)}(t). \end{aligned} \tag{35}$$

We deem that $a(z)$ is an unknown function and the function $g(z)$ is to be known.

Function $S(z)$ is a solution to Volterra integral equation of the second kind:

$$S(z) = \frac{1}{2}\gamma_0 - \frac{1}{2}\int_0^z a(\xi)\, S(\xi)\, d\xi, \quad z \in (0,l). \tag{36}$$

Inverting the operator $(\partial^2/\partial t^2) - (\partial^2/\partial z^2)$ in (31) and taking into account (33) and (34), we obtain

$$v(z,t) = \Phi(z,t) + A_{t,z}[Pv], \quad (z,t) \in \Delta(l). \tag{37}$$

Here we use the following notations:

$$\begin{aligned} \Phi(z,t) &= \frac{1}{2}\left[f(t+z) + f(t-z)\right] + \frac{1}{2}\int_{t-z}^{t+z} \varphi(\xi)\, d\xi, \\ A_{t,z}[v] &= \frac{1}{2}\int_0^z \int_{t-z+\xi}^{t+z-\xi} v(\xi,\tau)\, d\tau\, d\xi. \end{aligned} \tag{38}$$

Differentiating (37) with respect to t we obtain

$$\begin{aligned} &v_t(z,t) \\ &= \Phi_t(z,t) + \frac{1}{2}\int_0^z \left[Pv(\xi, t+z-\xi) - Pv(\xi, t-z+\xi)\right] d\xi. \end{aligned} \tag{39}$$

Put $t = z + 0$ in (37) and use condition (32); then we have

$$S(z) = \Phi(z, z+0) + A_{z+0,z}[Pv]. \tag{40}$$

Differentiating both sides of the resulting equality with respect to z gives

$$S'(z) = \Phi'(z, z+0) + \int_0^z Pv(\xi, 2z-\xi)\, d\xi. \tag{41}$$

It is not difficult to see that the function $q(z) = [S(z)]^{-1}$ satisfies Volterra integral equation of the second kind:

$$q(z) = \gamma^{-1} + \frac{1}{2}\int_0^z a(\xi)\, q(\xi)\, d\xi, \qquad \gamma = \frac{\gamma_0}{2}. \tag{42}$$

Taking this into account and the relation $a(z) = 2S'(z)/S(z)$, we get

$$\begin{aligned} a(z) = 2 &\left[\Phi'(z, z+0) + \int_0^z Pv(\xi, 2z-\xi)\, d\xi\right] \\ &\cdot \left[\gamma^{-1} + \frac{1}{2}\int_0^z a(\xi)\, q(\xi)\, d\xi\right]. \end{aligned} \tag{43}$$

Thus, we obtain a closed system of integral equations (37), (39), (42), and (43).

We write this system in vector form as follows:

$$\Upsilon = F + K(\Upsilon), \tag{44}$$

where

$$\begin{aligned} \Upsilon(z,t) &= \left(\Upsilon_1, \Upsilon_2, \Upsilon_3, \Upsilon_4\right)^T, \\ F(z,t) &= \left(F_1, F_2, F_3, F_4\right)^T, \\ K(\Upsilon) &= \left(K_1(\Upsilon), K_2(\Upsilon), K_3(\Upsilon), K_4(\Upsilon)\right)^T, \\ \Upsilon_1(z,t) &= v(z,t), \qquad \Upsilon_2(z,t) = v_t(z,t), \\ \Upsilon_3(z) &= q(z), \qquad \Upsilon_4(z) = a(z). \\ F_1(z,t) &= \Phi(z,t), \qquad F_2(z,t) = \Phi_t(z,t), \\ F_3 &= {\gamma_0}^{-1}, \qquad F_4(z) = \chi(z), \end{aligned} \tag{45}$$

where $\chi(z) = 2\gamma^{-1}\Phi'(z, z+0)$,

$$K_1(\Upsilon) = \frac{1}{2}\int_0^z \int_{t-z+\xi}^{t+z-\xi} P\Upsilon(\xi,\tau)\,d\tau\,d\xi, \quad (z,t) \in \Delta(l),$$

$$K_2(\Upsilon) = \frac{1}{2}\int_0^z \left[P\Upsilon(\xi, t+z-\xi) - P\Upsilon(\xi, t-z+\xi)\right] d\xi,$$

$$K_3(\Upsilon) = \frac{1}{2}\int_0^z \Upsilon_4(\xi)\,\Upsilon_3(\xi)\,d\xi,$$

$$K_4(\Upsilon) = \Phi'(z, z+0)\int_0^z \Upsilon_4(\xi)\,\Upsilon_3(\xi)\,d\xi + 2\int_0^z P\Upsilon(\xi, 2z-\xi)\,d\xi \cdot \left(\gamma^{-1} + \frac{1}{2}\int_0^z \Upsilon_4(\xi)\,\Upsilon_3(\xi)\,d\xi\right). \tag{46}$$

Here

$$P\Upsilon(z,t) = \Upsilon_4(z)\cdot\Upsilon_2(z,t) - g(z)\,\Upsilon_1(z,t). \tag{47}$$

We deem that $\Upsilon = (\Upsilon_1, \Upsilon_2, \Upsilon_3, \Upsilon_4) \in L_2(l)$, if

$$\Upsilon_j(z,t) \in L_2(\Delta(l)), \quad j = 1,2; \qquad \Upsilon_j(z) \in L_2(0,l), \quad j = 3,4. \tag{48}$$

Let $\Upsilon^{(j)} = (\Upsilon_1^{(j)}(z,t), \Upsilon_2^{(j)}(z,t), \Upsilon_3^{(j)}(z), \Upsilon_4^{(j)}(z))^T$, $j = 1,2$. We define the scalar product and the norm as follows:

$$\left\langle \Upsilon^{(1)}, \Upsilon^{(2)} \right\rangle = \sum_{k=1}^{2}\int_0^l \int_z^{2l-z} \Upsilon_k^{(1)}(z,t)\,\Upsilon_k^{(2)}(z,t)\,dt\,dz + \sum_{k=3}^{4}\int_0^l \Upsilon_k^{(1)}(z)\,\Upsilon_k^{(2)}(z)\,dz, \qquad \|\Upsilon\|^2 = \langle \Upsilon, \Upsilon\rangle. \tag{49}$$

Inverse Problem. Find vector $\Upsilon \in L_2(l)$ from (44) for given $F \in L_2(l)$.

4. Conditional Stability

Studying H_1, conditional stability is similar to that in [12] where it was done for the inverse acoustic problem.

We suppose $\|a\|^2_{L_2(0,l)} = M_1$, $\|f\|^2_{L_2(0,l)} + \|f'\|^2_{L_2(0,l)} = M_2$, $\|\varphi\|^2_{L_2(0,l)} = M_3$, and $\|g\|^2_{L_2(0,l)} \le M_4$ to be known.

We define $\sum(l, M_1, a_*)$ as the class of possible solutions of the inverse problem; namely, $a(z) \in \sum(l, M_1, a_*)$ if $a(z)$ satisfies the following conditions:

(1) $a(z) \in H_1(0,l) \cap C^1(0,l)$,

(2) $\|a\|_{H_{1(0,l)}} \le M_1$,

(3) $0 < a_* \le a(z), x \in (0,l)$.

We also define $F(l, M_2, M_3, M_4, k_0)$ as the class of possible initial data; namely, $f \in F(l, Q, k_0)$ if f satisfies the following conditions:

(1) $f \in H_1(0, 2l)$,

(2) $\|f\|_{H_1(0,2l)} \le M_2$,

(3) $f(+0) = k_0$, $\|\varphi\|_{H_1(0,2l)} \le M_3$.

Suppose that for $f^{(1)}, f^{(2)} \in F(l, M_2, M_3, M_4, k_0)$ there exist $a^{(1)}$ and $a^{(2)}$ from $\sum(l, M_1, a_*)$ which solve the inverse problem:

$$v_{tt}^{(j)}(z,t) = v_{zz}^{(j)}(z,t) - Pv^{(j)}(z,t), \quad (z,t) \in \Delta(l),$$
$$v^{(j)}(z,z) = S^{(j)}(z), \quad 0 \le z \le l,$$
$$v^{(j)}(0,t) = f^{(j)}(t), \qquad v_z(0,t) = \varphi^{(j)}(t), \quad 0 \le t \le 2l, \tag{50}$$

for $j = 1, 2$, respectively.

Here

$$Pv^{(j)}(z,t) = a^{(j)}(z)\,v_t^{(j)}(z,t) + g(z)\,v^{(j)}(z,t), \qquad f^{(j)}(t) = f_{(1)}^{(j)}(t), \qquad \varphi^{(j)}(t) = \mu f_{(2)}^{(j)}(t). \tag{51}$$

We deem that the function $g(z)$ is known and $a^{(j)}(z)$ is unknown, $j = 1, 2$. We write the early resulting closed system in the vector form as follows:

$$\Upsilon^{(j)} = F^{(j)} + K\left(\Upsilon^{(j)}\right), \quad j = 1,2, \tag{52}$$

where

$$\Upsilon^{(j)} = \left(\Upsilon_1^{(j)}, \Upsilon_2^{(j)}, \Upsilon_3^{(j)}, \Upsilon_4^{(j)}\right)^T; \tag{53}$$

$$\Upsilon_1^{(j)}(z,t) = v^{(j)}(z,t), \qquad \Upsilon_2^{(j)}(z,t) = v_t^{(j)}(z,t), \qquad \Upsilon_3^{(j)}(z) = q(z), \qquad \Upsilon_4^{(j)}(z) = a^{(j)}(z);$$

$$F^{(j)} = \left(F_1^{(j)}, F_2^{(j)}, F_3^{(j)}, F_4^{(j)}\right)^T, \quad j = 1,2; \qquad F_1^{(j)}(z,t) = \Phi^{(j)}(z,t), \qquad F_2^{(j)}(z,t) = \Phi_t^{(j)}(z,t), \qquad F_3 = j^{-1}, \qquad F_4(z) = \chi^{(j)}(z), \quad j = 1,2; \tag{54}$$

$$K_1\left(\Upsilon^{(j)}\right) = \frac{1}{2}\int_0^z \int_{t-z+\xi}^{t+z-\xi} P\Upsilon^{(j)}(\xi,\tau)\,d\tau\,d\xi, \quad (z,l) \in \Delta(l),$$

$$K_2\left(\Upsilon^{(j)}\right) = \frac{1}{2}\int_0^z \left[P\Upsilon^{(j)}(\xi, t+z-\xi) - P\Upsilon^{(j)}(\xi, t-z+\xi)\right] d\xi,$$

$$K_3\left(\Upsilon^{(j)}\right) = \frac{1}{2}\int_0^z \Upsilon_4^{(j)}\Upsilon_3^{(j)} d\xi,$$

$$K_4\left(\Upsilon^{(j)}\right) = \Phi'^{(j)}(z, z+0)\int_0^z \Upsilon_4^{(j)}(\xi)\,\Upsilon_3^{(j)}(\xi)\,d\xi + 2\int_0^z P\Upsilon^{(j)}(\xi, 2z-\xi)\,d\xi \cdot \left(\gamma^{-1} + \frac{1}{2}\int_0^z \Upsilon_4^{(j)}(\xi)\,\Upsilon_3^{(j)}(\xi)\,d\xi\right). \quad labeleq55 \tag{55}$$

Here we denote $P\Upsilon^{(j)}(z,t) = \Upsilon_4(z)\Upsilon_2(z,t) - g(z)\Upsilon_1(z,t)$.

Theorem 1. *Suppose that, for $F^{(j)} \in L_2(l)$, $j = 1, 2$, there exist $\Upsilon^{(j)} \in L_2(\Delta(l))$ as the solution of the inverse problem as follows:*

$$\Upsilon^{(j)}(z,t) = F^{(j)}(z,t) + K\left(\Upsilon^{(j)}\right), \quad j = 1,2,\ (z,t) \in \Delta(l). \tag{56}$$

Then

$$\left\|\Upsilon^{(1)} - \Upsilon^{(2)}\right\|^2 \le C\left\|f^{(1)} - f^{(2)}\right\|^2_{H_1(0,2l)}, \tag{57}$$

where

$$C = C\left(l, M_1, M_2, M_3, M_4, k_0\right). \tag{58}$$

Proof. We introduce

$$\widetilde{\Upsilon}(x,t) = \left(\widetilde{\Upsilon}_1(z,t), \widetilde{\Upsilon}_2(z,t), \widetilde{\Upsilon}_3(z), \widetilde{\Upsilon}_4(z)\right) = \Upsilon^{(1)}(z,t) - \Upsilon^{(2)}(z,t), \qquad \widetilde{F}(z,t) = F^{(1)}(z,t) - F^{(2)}(z,t). \tag{59}$$

Then from (52) it follows that

$$\widetilde{\Upsilon}(z,t) = \widetilde{F}(z,t) - K\left(\widetilde{\Upsilon}\right), \quad (z,t) \in \Delta(l). \tag{60}$$

In the vector equation (60) we estimate each component separately taking into account the obvious inequalities as follows:

$$(a+b+c)^2 \le 3\left(a^2+b^2+c^2\right), \qquad \left(\sqrt{a}+\sqrt{b}\right)^2 \le 2a+2b, \tag{61}$$

for $a \ge 0, b \ge 0$.

We obtain the chain of the inequalities:

$$\begin{aligned}\left|\widetilde{\Upsilon}_1(z,t)\right| &\le \left|\widetilde{F}_1(z,t)\right| + \frac{1}{2}\sqrt{\int_0^z \left|\widetilde{\Upsilon}_3(\xi)\right|^2 d\xi} \\ &\times \left[\sqrt{\int_0^z \left|\Upsilon_1^{(1)}(\xi, t+z-\xi)\right|^2 d\xi} + \sqrt{\int_0^z \left|\Upsilon_1^{(1)}(\xi, t-z+\xi)\right|^2 d\xi}\right] \\ &+ \frac{1}{2}\sqrt{\int_0^z \left|\Upsilon_3^{(2)}(\xi)\right|^2 d\xi} \\ &\times \left[\sqrt{\int_0^z \left|\widetilde{\Upsilon}_1(\xi, t+z-\xi)\right|^2 d\xi} + \sqrt{\int_0^z \left|\widetilde{\Upsilon}_2(\xi, t-z+\xi)\right|^2 d\xi}\right].\end{aligned} \tag{62}$$

Using the obvious inequality we get

$$\begin{aligned}\left|\widetilde{\Upsilon}_1(z,t)\right|^2 &\le 3\left|\widetilde{F}_1(z,t)\right|^2 + \frac{3}{2}\int_0^z \left|\widetilde{\Upsilon}_3(\xi)\right|^2 d\xi \\ &\times \int_0^z \left[\left|\Upsilon_1^{(1)}(\xi, t+z-\xi)\right|^2 + \left|\Upsilon_1^{(1)}(\xi, t-z+\xi)\right|^2\right] d\xi \\ &+ \frac{3}{2}\int_0^z \left|\Upsilon_3^{(2)}(\xi)\right|^2 d\xi \\ &\times \int_0^z \left|\widetilde{\Upsilon}_1(\xi, t+z-\xi)\right|^2 + \left|\widetilde{\Upsilon}_1^{(2)}(\xi, t-z+\xi)\right|^2 d\xi.\end{aligned} \tag{63}$$

Turning to the earlier introduced norms we have

$$\begin{aligned}\left\|\widetilde{\Upsilon}_1\right\|^2_{L_2(\Delta(l,z))} &\le 3\left\|\widetilde{F}_1\right\|^2_{L_2(\Delta(l,z))^T} \\ &+ \frac{3}{2}\int_0^z \int_\xi^{2l-\xi} \left\{\int_0^\xi \left|\widetilde{\Upsilon}_3\left(\xi'\right)\right|^2 d\xi' \right. \\ &\left. \times \int_0^\xi \left[\left|\Upsilon_1^{(1)}\left(\xi', \tau+\xi-\xi'\right)\right|^2\right.\right.\end{aligned}$$

$$+\left|\Upsilon_1^{(1)}\left(\xi', \tau-\xi+\xi'\right)\right|^2\Big] d\xi' + \int_0^{\xi}\left|\Upsilon_3^2\left(\xi'\right)\right|^2 d\xi' \times \int_0^{\xi}\Big[\left|\widetilde{\Upsilon}_1\left(\xi', \tau+\xi-\xi'\right)\right|^2 + \left|\widetilde{\Upsilon}_1^{(2)}\left(\xi', \tau-\xi+\xi'\right)\right|^2\Big] d\xi'\Bigg\} d\tau\, d\xi$$

$$\leq 3\left\|\widetilde{F}_1\right\|^2_{L_2(\Delta(l,z))} + 12\Upsilon_*^2\int_0^z\int_0^{\xi}\left|\widetilde{\Upsilon}_3\left(\xi'\right)\right|^2 d\xi'\, d\xi + 12\Upsilon_*^2\int_0^z\left\|\widetilde{\Upsilon}_1\right\|_{L_2(\Delta(l,\xi))} d\xi. \tag{64}$$

Here

$$\Upsilon_* = \max\left\{\left\|\Upsilon^{(1)}\right\|, \left\|\Upsilon^{(2)}\right\|\right\}. \tag{65}$$

We estimate the second component of (60):

$$\begin{aligned}\left|\widetilde{\Upsilon}_2(z)\right| &\leq \frac{1}{2}\int_0^z\left|\Upsilon_3^{(1)}(\xi)\,\widetilde{\Upsilon}_2(\xi)\right| d\xi + \frac{1}{2}\int_0^z\left|\Upsilon_2^{(2)}(\xi)\,\widetilde{\Upsilon}_3(\xi)\right| d\xi \\ &\leq \frac{1}{2}\sqrt{\int_0^z\left|\Upsilon_3^{(1)}(\xi)\right|^2 d\xi}\sqrt{\int_0^z\left|\widetilde{\Upsilon}_2(\xi)\right|^2 d\xi} + \frac{1}{2}\sqrt{\int_0^z\left|\Upsilon_2^{(2)}(\xi)\right|^2 d\xi}\sqrt{\int_0^z\left|\widetilde{\Upsilon}_3(\xi)\right|^2 d\xi}.\end{aligned} \tag{66}$$

Then we have

$$\left\|\widetilde{\Upsilon}_2\right\|^2_{L_2(0,z)} \leq \frac{1}{2}\Upsilon_*^2\int_0^z\left[\left\|\widetilde{\Upsilon}_2\right\|^2_{L_2(0,\xi)} + \left\|\widetilde{\Upsilon}_3\right\|^2_{L_2(0,\xi)}\right] d\xi. \tag{67}$$

We estimate the third component of (60) and we have

$$\left\|\widetilde{\Upsilon}_3\right\|^2_{L_2(0,l)} \leq \frac{1}{4}M_2\int_0^z\left[\left\|\widetilde{\Upsilon}_3\right\|^2_{L_2(0,\xi)} + M_3\left\|\widetilde{\Upsilon}_4\right\|^2_{L_2(0,\xi)}\right] d\xi. \tag{68}$$

Finally, for the fourth component of (60) we get the estimate

$$\begin{aligned}\left|\widetilde{\Upsilon}_4(z)\right| &\leq \left|\widetilde{F}_4(z)\right| + \sum_{i=1}^4 w_i,\\ w_1(z) &= 2\left|\left(f^{(1)}\right)'(z)\right|\left|K_2\left(\Upsilon^{(1)}\right) - K_2\left(\Upsilon^{(2)}\right)\right|,\\ w_2(z) &= \left|K_6\left(\Upsilon^{(1)}\right) - K_6\left(\Upsilon^{(2)}\right)\right|,\\ w_3 &= \left|K_2\left(\Upsilon^{(1)}\right)\right| w_2(z),\\ w_4 &= \left|K_4\left(\Upsilon^{(2)}\right)\right|\left|K_2\left(\Upsilon^{(1)}\right) - K_2\left(\Upsilon^{(2)}\right)\right|,\\ w_5 &= \left|\left(f^{(1)}\right)' - \left(f^{(2)}\right)\right|\left|K_2\left(\Upsilon^{(2)}\right)\right|,\\ K_6(\Upsilon) &= \int_0^z \Upsilon_3(\xi)\,\Phi_1(\xi, 2z-\xi)\, d\xi.\end{aligned} \tag{69}$$

Estimating each term $w_i(z)$ and substituting into (69) and using the obvious inequality

$$\left(\sum_{k=1}^4 |b_k|\right)^2 \leq 4\sum_{k=1}^4 |b_k|^2, \tag{70}$$

we obtain

$$\begin{aligned}\left\|\widetilde{\Upsilon}_4(z)\right\|_{L_2(0,z)} &\leq \nu_0\int_0^z\left[f'(2\xi)\right]^2 d\xi + \frac{1}{2}\nu_1\left\|\widetilde{\Upsilon}_1\right\|_{L_2(\Delta(l,z))} + \int_0^z \nu_3(\xi)\left\|\widetilde{\Upsilon}_2\right\|_{L_2(0,\xi)} d\xi \\ &+ \int_0^z \nu_3(\xi)\left\|\widetilde{\Upsilon}_3\right\|_{L_2(0,\xi)} d\xi + \int_0^z \nu_4(\xi)\left\|\widetilde{\Upsilon}_4\right\|_{L_2(0,\xi)} d\xi.\end{aligned} \tag{71}$$

Now we combine all the obtained estimates for the four components (60) and denote, for convenience,

$$\psi_1(z) = \left\|\widetilde{\Upsilon}_1\right\|^2_{L_2(\Delta(l,z))}, \quad z \in (0, l), \tag{72}$$

and then

$$\psi(z) = \psi_1(z) + \psi_2(z) + \psi_3(z) + \psi_4(z) \tag{73}$$

and for function ψ we obtain the following estimate:

$$\psi(z) \leq \eta + \int_0^z\sum_{i=1}^4 \gamma_i(\xi)\,\psi_i(\xi)\, d\xi, \tag{74}$$

where $\eta = \eta(\Upsilon_*^2, \nu_1, \nu_2, \nu_3, \nu_4)$.

Introduce a new function:

$$\nu(z) = \eta_* + \int_0^z\sum_{i=1}^4 \gamma_i(\xi)\,\psi_i(\xi)\, d\xi, \quad \eta < \eta_*, \tag{75}$$

where η_* is constant.

Then $\psi(z) \leq \nu(z)$,

$$\begin{aligned}\nu'(z) &= \sum_{i=1}^4 \gamma_i(z)\,\Upsilon_j(z) \leq \nu(z)\sum_{i=1}^4 \gamma_i(z),\\ \frac{\nu'(z)}{\nu(z)} &\leq \sum_{i=1}^4 \gamma_i(z).\end{aligned} \tag{76}$$

Applying the Gronwall inequality we obtain

$$\psi(z) \le \nu(z) \le \nu(0) \exp\left\{\int_0^z \sum_{i=1}^4 \gamma_i(\xi)\, d\xi\right\},$$

$$\int_0^z \sum_{i=1}^4 \gamma_i(\xi)\, d\xi \le 25\Upsilon_*^2 \times z + 12\Upsilon_*^2 \left\|f^{(1)}\right\|^2_{L_2(0,2l)} + 12\Upsilon_*^4 + 12\Upsilon_*^2\left(12 + \Upsilon_*^2 \cdot z\right). \tag{77}$$

Then from (77) we obtain

$$\left\|\Upsilon^{(1)} - \Upsilon^{(2)}\right\|^2 \le \widetilde{N}\left\|f^{(1)} - f^{(2)}\right\|_{H_1(0,2l)}, \tag{78}$$

where the constant $C > 0$ is given by (58).

An explicit expression for the constant as a result of successive computations is given by

$$C = \left[6l + 6M_1\left(\frac{4}{k_0^2} + \Upsilon_*^4\right)\left(1 + 12\Upsilon_*^2 l\right)\right] \times \exp\left\{\Upsilon_*^2\left[24l + 8M_2\left(\frac{4}{k_0^2} + \Upsilon_*^4\right)\left(M_3 + 36\Upsilon_*^2 l\right) + 6M_2\Phi^2 + 8M_4\Upsilon_*^4\right]\right\}. \tag{79}$$

□

5. Conclusions

The conditional stability of the inverse problem for the geoelectric equation has been investigated. For studying we consider the integral formulation of the inverse geoelectric problem. The estimation of the conditional stability of the inverse problem solution has been obtained or rather lower changes in input data imply lower changes in the solution (of the numerical method). When determining the additional information the device errors are possible. That is why this research is important for experimental studies with usage of ground penetrating radars. The inlet data belongs to the class $F(l, M_2, M_3, M_4, k_0)$, while the solution belongs to the class $\sum(l, M_1, a_*)$.

Conflict of Interests

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

Acknowledgment

The work was supported by Ministry of Education and Science of the Republic of Kazakhstan (Grant no. 139 (69) от 04.02.2014).

References

[1] S. I. Kabanikhin, *Inverse and III-Posed Problems: Theory and Applications*, de Gruyter, Berlin, Germany, 2011.

[2] A. S. Blagoveshchenskii and V. M. Babič, "On a local method of solution of a nonstationary inverse problem for a nonhomogeneous string," in *Mathematical Questions in the Theory of Wave Diffraction*, vol. 115 of *Proceedings of the Steklov Institute of Mathematics in the Academy of Sciences of the USSR*, pp. 30–41, American Mathematical Society, Providence, RI, USA, 1974.

[3] V. G. Romanov, *Inverse Problems of Mathematical Physics*, VNU Science Press, Utrecht, The Netherlands, 1987.

[4] S. I. Kabanikhin and A. Lorenzi, *Identification Problems of Wave phenomena*, VSP, Utrecht, The Netherlands, 1999.

[5] V. G. Romanov and M. Yamamoto, "Multidimensional inverse hyperbolic problem with impulse input and a single boundary measurement," *Journal of Inverse and Ill-Posed Problems*, vol. 7, no. 6, pp. 573–588, 1999.

[6] V. Baranov and G. Kunetz, "Synthetic seismograms with multiplenreflections: theory and numerical experience," *Geophysical Prospecting*, vol. 8, pp. 315–325, 1969.

[7] A. Bamberger, G. Chavent, C. Hemon, and P. Lailly, "Inversion of normal incidence seismograms," *Geophysics*, vol. 47, no. 5, pp. 757–770, 1982.

[8] A. Bamberger, G. Chavent, and P. Lailly, "About the stability of the inverse problem in 1-D wave equations—applications to the interpretation of seismic profiles," *Applied Mathematics and Optimization*, vol. 5, no. 1, pp. 1–47, 1979.

[9] S. He and S. I. Kabanikhin, "An optimization approach to a three-dimensional acoustic inverse problem in the time domain," *Journal of Mathematical Physics*, vol. 36, no. 8, pp. 4028–4043, 1995.

[10] J. S. Azamatov and S. I. Kabanikhin, "Volterra operator equations. L_2-theory," *Journal of Inverse and Ill-Posed Problems*, vol. 7, no. 6, pp. 487–510, 1999.

[11] V. G. Romanov and S. I. Kabanikhin, *Inverse Problems for Maxwell's Equations*, VSP, Utrecht, The Netherlands, 1994.

[12] S. I. Kabanikhin, K. T. Iskakov, and M. Yamamoto, "H^1-conditional stability with explicit Lipshitz constant for a one-dimensional inverse acoustic problem," *Journal of Inverse and Ill-Posed Problems*, vol. 9, no. 3, pp. 249–267, 2001.

[13] S. I. Kabanikhin and K. T. Iskakov, "Justification of the steepest descent method in an integral formulation of an inverse problem for a hyperbolic equation," *Siberian Mathematical journal*, vol. 42, no. 3, pp. 478–494, 2001.

[14] S. I. Kabanikhin and K. T. Iskakov, *Optimization Methods of Coefficient Inverse Problems Solution*, NGU, Novosibirsk, Russia, 2001.

[15] A. L. Bukhgeim and M. V. Klibanov, "Uniqueness in the large of a class of multidimensional inverse problems," *Soviet Mathematics Doklady*, vol. 17, pp. 244–247, 1981.

[16] M. V. Klibanov, "Carleman estimates for global uniqueness, stability and numerical methods for coefficient inverse problems," *Journal of Inverse and Ill-Posed Problems*, vol. 21, no. 4, pp. 477–560, 2013.

[17] M. V. Klibanov, "Uniqueness of the solution of two inverse problems for a Maxwell system," *Computational Mathematics and Mathematical Physics*, vol. 26, no. 7, pp. 67–73, 1986.

[18] A. V. Kuzhuget, L. Beilina, M. V. Klibanov, A. Sullivan, L. Nguyen, and M. A. Fiddy, "Blind backscattering experimental data collected in the field and an approximately globally convergent inverse algorithm," *Inverse Problems*, vol. 28, no. 9, Article ID 095007, 2012.

[19] L. Beilina and M. V. Klibanov, *Approximate Global Convergence and Adaptivity for Coefficient Inverse Problems*, Springer, New York, NY, USA, 2012.

[20] A. L. Karchevsky, M. V. Klibanov, L. Nguyen, N. Pantong, and A. Sullivan, "The Krein method and the globally convergent method for experimental data," *Applied Numerical Mathematics*, vol. 74, pp. 111–127, 2013.

[21] S. I. Kabanikhin, D. B. Nurseitov, M. A. Shishlenin, and B. B. Sholpanbaev, "Inverse problems for the ground penetrating radar," *Journal of Inverse and Ill-Posed Problems*, vol. 21, no. 6, pp. 885–892, 2013.

Hindawi Publishing Corporation
Mathematical Problems in Engineering
Volume 2014, Article ID 157893, 11 pages
http://dx.doi.org/10.1155/2014/157893

Research Article

An Adaptive Total Generalized Variation Model with Augmented Lagrangian Method for Image Denoising

Chuan He,[1] Changhua Hu,[1] Xiaogang Yang,[2] Huafeng He,[1] and Qi Zhang[1]

[1] *Unit 302, Xi'an Institute of High-Tech, Xi'an 710025, China*
[2] *Unit 303, Xi'an Institute of High-Tech, Xi'an 710025, China*

Correspondence should be addressed to Chuan He; hechuan8512@163.com

Received 3 March 2014; Revised 20 May 2014; Accepted 25 May 2014; Published 10 July 2014

Academic Editor: Fatih Yaman

We propose an adaptive total generalized variation (TGV) based model, aiming at achieving a balance between edge preservation and region smoothness for image denoising. The variable splitting (VS) and the classical augmented Lagrangian method (ALM) are used to solve the proposed model. With the proposed adaptive model and ALM, the regularization parameter, which balances the data fidelity and the regularizer, is refreshed with a closed form in each iterate, and the image denoising can be accomplished without manual interference. Numerical results indicate that our method is effective in staircasing effect suppression and holds superiority over some other state-of-the-art methods both in quantitative and in qualitative assessment.

1. Introduction

In the past few decades, many variation or partial differential equation (PDE) based restoration models [1–7] have been proposed to recover images from degraded observations, due to the ability of preserving significant image features such as edges or textures. Among these models, the total variation (TV) model, also named the Rudin-Osher-Fatemi (ROF) model [1], is distinguished for excellent edge preserving ability and becomes one of the most widely used regularizers in image restoration [1, 2, 8–10]. In particular, the TV denoising problem is in the following form:

$$\operatorname*{argmin}_{u} \int_{\Omega} |\nabla u| \, d\mathbf{x} + \frac{\lambda}{2} \int_{\Omega} |u - u_0|^2 d\mathbf{x}, \tag{1}$$

where Ω is an open bounded domain in two dimensions, u is the image to be restored, u_0 is the observation containing Gaussian white noise, and λ is the regularization parameter which balances the regularization term and the data fidelity term. $\int_{\Omega} |\nabla u| d\mathbf{x}$ is the TV seminorm of the bounded variation (BV) space BV(Ω). The TV model is highly effective in preserving edges and corners, compared with the quadratic Tikhonov model. However, only when the original image is piecewise constant, the TV model is proved to be optimal. In fact, staircasing effect usually appears because most of natural images are not piecewise constant. Staircasing effect cannot meet the demands of human vision, due to the new artificial edges which do not exist in original images.

To overcome the drawback of the TV model, researchers suggest introducing the higher-order derivatives of image functions [3–7, 12–16]. In order to eliminate the staircasing effect of TV model, Chambolle and Lions [14] proposed the following infimal-convolution minimization functional:

$$\begin{aligned} \operatorname*{argmin}_{u,v} & \int_{\Omega} |\nabla u - \nabla v| \, d\mathbf{x} + \alpha \int_{\Omega} \left|\partial^2 v\right| d\mathbf{x} \\ & + \lambda \int_{\Omega} |u - u_0|^2 d\mathbf{x}, \end{aligned} \tag{2}$$

where discontinuous components of the image are allotted to $u - v$ while regions of moderate slopes are assigned to v. The above model was proved to be practically efficient. Later, a modified form of (2) was proposed in [5] and its regularizer is of the following form:

$$\operatorname*{argmin}_{u,v} \int_{\Omega} |\nabla u - \nabla v| \, d\mathbf{x} + \alpha \int_{\Omega} |\Delta v| \, d\mathbf{x}. \tag{3}$$

That is, the second-order derivative in (2) is substituted by the Laplacian in (3). The similar use of Laplacian operator can also be seen in some PDE-based methods [3].

Since the classical TV model could not distinguish jumps from smooth transitions, Chan et al. [12] considered an additional penalization of the discontinuities in images. Precisely, they adopt

$$\int_\Omega |\nabla u| \, \mathrm{d}\mathbf{x} + \alpha \int_\Omega \psi\left(|\nabla u|\right) h\left(\Delta u\right) \mathrm{d}\mathbf{x} \tag{4}$$

as the regularization term, where ψ is a real-valued function whose value approaches 0 while $|\nabla u|$ approaches infinity. The absence of the staircasing effect for this choice was verified in [15].

Bredies et al. [17] proposed the concept of total generalized variation (TGV), which is considered to be the generalization of TV. The TGV model is defined as

$$\begin{aligned}\mathrm{TGV}_{\boldsymbol{\alpha}}^k(u) = \sup\Big\{ \int_\Omega u \ \mathrm{div}^k v \ \mathrm{d}\mathbf{x} \mid v \in C_c^k\left(\Omega, \mathrm{Sym}^k\left(\mathbb{R}^d\right)\right), \\ \left\|\mathrm{div}^l v\right\|_\infty \le \alpha_l, \\ l = 0, \ldots, k-1 \Big\}\end{aligned}$$

with

$$\begin{aligned}\mathrm{Sym}^k\left(\mathbb{R}^d\right) = \Big\{ \zeta : \underbrace{\mathbb{R}^d \times \mathbb{R}^d \cdots \times \mathbb{R}^d}_{k \text{ times}} \longrightarrow \mathbb{R} \\ \mid \zeta \text{ is } k\text{-linear and symmetric} \Big\},\end{aligned} \tag{5}$$

where $d \ge 1$ denotes the image dimension, and, throughout this paper, we assume $d = 2$; $\mathrm{Sym}^k(\mathbb{R}^d)$ is the space of symmetric k-tensors on $\mathbb{R}^d$; $C_c^k(\Omega, \mathrm{Sym}^k(\mathbb{R}^d))$ is the space of compactly supported symmetric tensor field; α_l is fixed positive parameter. From the definition of $\mathrm{TGV}_{\boldsymbol{\alpha}}^k$, we learn that it involves the derivatives of u of order one to k. When $k = 1$ and $\alpha_0 = 1$, $\mathrm{TGV}_{\boldsymbol{\alpha}}^k$ degenerates to the classical TV. Thus TGV can be seen as a generalization of TV.

TGV involves and balances higher-order derivatives of u. Image reconstruction with TGV regularization usually leads to result with piecewise polynomial intensities and sharp edges. Therefore, TGV can effectively suppress the staircasing effect. In [17], an accelerated first-order method of Nesterov [18] was proposed to solve the TGV-regularized denoising problem.

In this paper, we propose an adaptive second-order TGV-regularized model for denoising and derive an augmented Lagrangian approach to handle the suggested model. Our denoising model is as follows:

$$\underset{u}{\mathrm{argmin}} \mathrm{TGV}_{\boldsymbol{\alpha}}^2(u) \quad \text{s.t.} \quad \int_\Omega \left|u - u_0\right|^2 \mathrm{d}\mathbf{x} \le c. \tag{6}$$

According to the standard Lagrange duality, for a given c, there exists a nonnegative λ such that

$$\underset{u}{\mathrm{argmin}} \mathrm{TGV}_{\boldsymbol{\alpha}}^2(u) + \frac{\lambda}{2} \int_\Omega \left|u - u_0\right|^2 \mathrm{d}\mathbf{x} \tag{7}$$

is equivalent to (6). However, with (6), we can automatically estimate the regularization parameter λ. We first utilize an indicator function of the feasible set to transform problem (6) into an unconstrained one; then the variable splitting technique is applied to transform the resulting unconstrained problem into a problem with linear penalizing constraints; finally, the obtained constrained problem is solved by the alternating direction method of multipliers (ADMM) [19–22], which is an instance of the classical ALM. The resulting image denoising algorithm is effective in staircasing effect suppression compared with some TV-based denoising methods, due to the second-order TGV regularizer. Besides, it achieves the adaptive estimation of the regularization parameter without inner iterative scheme. It is worth noting that the idea of this paper can be extended to TGV models with higher order than two. However, for simplicity, we only treat the second-order model and this is adequate for a large class of natural images.

Our method differs from the previous works on at least two aspects. On one hand, compared with [16], which adopted the accelerated first-order method of Nesterov [18] to handle the unconstrained TGV-based denoising problem (7), we apply ALM to the constrained TGV-based denoising problem (6) and achieve the automatic estimation of the regularization parameter λ. Our strategy avoids the extra cost on the manual selection of λ by try-and-error. On the other hand, compared with the existing TV-based adaptive methods [10, 23, 24], we propose a more complicated adaptive method based on TGV, and it is apt to achieve more attractive results than the TV-based methods.

The outline of the rest of the paper is organized as follows. Section 2 provides the description of the adaptive second-order TGV-based model for image denoising. Based on the Lagrange duality, an equivalent form of $\mathrm{TGV}_{\boldsymbol{\alpha}}^2$ is suggested. The derivation of the proposed method is presented in Section 3. Section 4 gives the numerical results that demonstrate the effectiveness of the proposed method. At last, Section 5 ends this paper with a brief conclusion.

2. Adaptive Second-Order TGV-Based Model for Image Denoising

The space of bounded generalized variation (BGV) functions of order k with weight $\boldsymbol{\alpha}$ is defined as

$$\mathrm{BGV}_\alpha^k(\Omega) = \left\{ u \in L^1(\Omega) \,\middle|\, \mathrm{TGV}_{\boldsymbol{\alpha}}^k(u) < \infty \right\}. \tag{8}$$

Correspondingly, the BGV norm is defined as

$$\|u\|_{\mathrm{BGV}_\alpha^k} = \|u\|_1 + \mathrm{TGV}_{\boldsymbol{\alpha}}^k(u). \tag{9}$$

The TGV seminorm rather than the BGV norm is usually used as a regularizer.

In this paper, we just take $k = 2$ into consideration for simplicity. The second-order TGV can be written as

$$\mathrm{TGV}^2_{\boldsymbol{\alpha}}(u) = \sup\left\{\int_\Omega u\ \mathrm{div}^2 v\ \mathrm{d}\mathbf{x}\,\middle|\, v \in C_c^2\left(\Omega, \mathrm{Sym}^2\left(\mathbb{R}^d\right)\right),\ \|v\|_\infty \le \alpha_0, \|\mathrm{div}\, v\|_\infty \le \alpha_1\right\}, \tag{10}$$

where the divergences are defined as

$$(\mathrm{div}\ v)_i = \sum_{j=1}^{d} \frac{\partial v_{ij}}{\partial x_j}, \quad 1 \le i \le d; \quad \mathrm{div}^2 v = \sum_{i,j=1}^{d} \frac{\partial^2 v_{ij}}{\partial x_i \partial x_j}. \tag{11}$$

In fact, $\mathrm{Sym}^2(\mathbb{R}^d)$ is equivalent to the space of all symmetric $d \times d$ matrices. The infinite norms in (10) are given by

$$\|v\|_\infty = \sup_{\mathbf{x}\in\Omega}\left(\sum_{i,j=1}^{d}\left|v_{ij}(\mathbf{x})\right|^2\right)^{1/2}, \qquad \|\mathrm{div}\ v\|_\infty = \sup_{\mathbf{x}\in\Omega}\left(\sum_{i=1}^{d}\left|(\mathrm{div}\ v)_i\right|^2\right)^{1/2}. \tag{12}$$

For the convenience of the derivation of our algorithm, we apply the discrete form in the following and the tensors and vectors are denoted in bold type font. In order to make use of ADMM, we apply an equivalent definition of $\mathrm{TGV}^2_{\boldsymbol{\alpha}}$ [17, 22] based on the Lagrange duality. With this definition, we have

$$\mathrm{TGV}^2_{\boldsymbol{\alpha}}(\mathbf{u}) = \min_{\mathbf{p}} \alpha_0 \|\varepsilon(\mathbf{p})\|_1 + \alpha_1 \|\nabla\mathbf{u} - \mathbf{p}\|_1, \tag{13}$$

where $\mathbf{u} \in \mathbb{R}^{mn}$ denotes an $m \times n$ image, $\mathbf{p} \in \mathbb{R}^{mn} \times \mathbb{R}^{mn}$ belongs to the two-dimensional 1-tensor field, and ε denotes the symmetrized derivative operator. Suppose that $u_{i,j}$ and $\mathbf{p}_{i,j}$ denote the (i, j)th components of $\mathbf{u}$ and $\mathbf{p}$, respectively. Then we have $\mathbf{p}_{i,j} = [p_{i,j,1}, p_{i,j,2}] \in \mathrm{Sym}^1(\mathbb{R}^2)$ and the (i, j)th component of $\varepsilon(\mathbf{p})$ is given by

$$\varepsilon(\mathbf{p})_{i,j} = \begin{bmatrix} \varepsilon(\mathbf{p})_{i,j,1} & \varepsilon(\mathbf{p})_{i,j,3} \\ \varepsilon(\mathbf{p})_{i,j,3} & \varepsilon(\mathbf{p})_{i,j,2} \end{bmatrix} = \begin{bmatrix} \nabla_{x_1} p_{i,j,1} & \dfrac{\left(\nabla_{x_2} p_{i,j,1} + \nabla_{x_1} p_{i,j,2}\right)}{2} \\ \dfrac{\left(\nabla_{x_2} p_{i,j,1} + \nabla_{x_1} p_{i,j,2}\right)}{2} & \nabla_{x_2} p_{i,j,2} \end{bmatrix} \in \mathrm{Sym}^2\left(\mathbb{R}^2\right) \quad 0 \le i \le m,\ 0 \le j \le n, \tag{14}$$

where ∇_{x1} and ∇_{x_2} denote the difference operators in directions x_1 and x_2. According to the definition of operators ∇ and ε, $(\nabla\mathbf{u})_{i,j}$ and $\varepsilon(\mathbf{p})_{i,j}$ are two-dimensional 1-tensor and symmetric 2-tensor, respectively. Besides, the $\|\|_1$s of $\mathbf{p}$ and $\varepsilon(\mathbf{p})$ are defined as $\|\mathbf{p}\|_1 = \sum_{i,j}^{m,n} |\mathbf{p}_{i,j}| = \sum_{i,j}^{m,n} \sqrt{p_{i,j,1}^2 + p_{i,j,2}^2}$ and $\|\varepsilon(\mathbf{p})\|_1 = \sum_{i,j}^{m,n} |\varepsilon(\mathbf{p})_{i,j}| = \sum_{i,j}^{m,n} \sqrt{\varepsilon(\mathbf{p})_{i,j,1}^2 + \varepsilon(\mathbf{p})_{i,j,2}^2 + 2\varepsilon(\mathbf{p})_{i,j,3}^2}$, respectively. The deduction of (13) is given in the Appendix.

Then the constrained second-order TGV-regularized denoising problem (6) can be rewritten as

$$\begin{aligned} \underset{\mathbf{u},\mathbf{p}}{\mathrm{argmin}} \quad & \alpha_0 \|\varepsilon(\mathbf{p})\|_1 + \alpha_1 \|\nabla\mathbf{u} - \mathbf{p}\|_1 \\ \text{s.t.} \quad & \left\|\mathbf{u} - \mathbf{u}_0\right\|_2^2 \le c. \end{aligned} \tag{15}$$

3. Methodology

3.1. The Augmented Lagrangian Model of Adaptive TGV-Based Denoising. Problem (15) can be transformed into an unconstrained problem, with the following discontinuous objective functional:

$$\underset{\mathbf{u},\mathbf{p}}{\mathrm{argmin}}\ \alpha_0 \|\varepsilon(\mathbf{p})\|_1 + \alpha_1 \|\nabla\mathbf{u} - \mathbf{p}\|_1 + I_\Phi(\mathbf{u}), \tag{16}$$

where $I_\Phi(\mathbf{u})$ is the indicator function of the feasible set defined by

$$I_\Phi(\mathbf{u}) = \begin{cases} 0, & \text{if } \mathbf{u} \in \Phi \triangleq \left\{\mathbf{u} : \left\|\mathbf{u} - \mathbf{u}_0\right\|_2^2 \le c\right\}; \\ +\infty & \text{otherwise.} \end{cases} \tag{17}$$

Note that Φ is a closed Euclidean ball centered at $\mathbf{u}_0$ with radius $\sqrt{c}$.

The solution of problem (16) suffers from its nonlinearity and nondifferentiability. Referring to the variable splitting, we introduce three auxiliary variables to simplify the solution process of (16): a variable $\mathbf{w}$ for liberating $\mathbf{u}$ out from the constraint of the feasible set; a variable $\mathbf{y}$ and a variable $\mathbf{z}$ for liberating $\varepsilon(\mathbf{p})$ and $\nabla\mathbf{u} - \mathbf{p}$ out from the nondifferentiable 1-norms, respectively. Then problem (16) can be transformed into the following equivalent constrained problem:

$$\begin{aligned} \underset{\mathbf{u},\mathbf{p}}{\mathrm{argmin}} \quad & \alpha_0 \|\mathbf{y}\|_1 + \alpha_1 \|\mathbf{z}\|_1 + I_\Phi(\mathbf{w}) \\ \text{s.t.} \quad & \mathbf{u} = \mathbf{w}, \quad \varepsilon(\mathbf{p}) = \mathbf{y}, \quad \nabla\mathbf{u} - \mathbf{p} = \mathbf{z}. \end{aligned} \tag{18}$$

In order to liberate $\mathbf{u}$ out from the feasible set constraint, we introduce auxiliary variable $\mathbf{w}$. Similar operation can also be found in [10]. Without this operation, we should resort to an inner iterative scheme to update the regularization parameter.

The corresponding augmented Lagrangian functional of (18) is defined as

$$\begin{aligned} \mathscr{L}_{\mathscr{A}}(\mathbf{u}, \mathbf{p}, \mathbf{w}, \mathbf{y}, \mathbf{z}; \boldsymbol{\mu}, \boldsymbol{\xi}, \boldsymbol{\eta}) \triangleq\ & I_\Phi(\mathbf{w}) - \langle \boldsymbol{\mu}, \mathbf{w} - \mathbf{u} \rangle \\ & + \frac{\beta_1}{2}\|\mathbf{w} - \mathbf{u}\|_2^2 + \alpha_0 \|\mathbf{y}\|_1 \\ & - \langle \boldsymbol{\xi}, \mathbf{y} - \varepsilon(\mathbf{p}) \rangle \end{aligned}$$

$$+\frac{\beta_2}{2}\|\mathbf{y}-\varepsilon(\mathbf{p})\|_2^2+\alpha_1\|\mathbf{z}\|_1 -\langle\boldsymbol{\eta},\mathbf{z}-\nabla\mathbf{u}+\mathbf{p}\rangle +\frac{\beta_3}{2}\|\mathbf{z}-\nabla\mathbf{u}+\mathbf{p}\|_2^2, \tag{19}$$

where $\boldsymbol{\mu}$, $\boldsymbol{\xi}$, and $\boldsymbol{\eta}$ are Lagrange multipliers and β_1, β_2, and β_3 are penalty parameters which should be positive. According to the classical ADMM, we should solve the following iterative scheme:

$$\begin{aligned}
\mathbf{u}^{k+1}&=\underset{\mathbf{u}}{\operatorname{argmin}}\,\mathscr{L}_{\mathscr{A}}\left(\mathbf{u},\mathbf{p}^k,\mathbf{w}^k,\mathbf{y}^k,\mathbf{z}^k;\boldsymbol{\mu}^k,\boldsymbol{\xi}^k,\boldsymbol{\eta}^k\right),\\
\mathbf{p}^{k+1}&=\underset{\mathbf{p}}{\operatorname{argmin}}\,\mathscr{L}_{\mathscr{A}}\left(\mathbf{u}^{k+1},\mathbf{p},\mathbf{w}^k,\mathbf{y}^k,\mathbf{z}^k;\boldsymbol{\mu}^k,\boldsymbol{\xi}^k,\boldsymbol{\eta}^k\right),\\
\mathbf{w}^{k+1}&=\underset{\mathbf{w}}{\operatorname{argmin}}\,\mathscr{L}_{\mathscr{A}}\left(\mathbf{u}^{k+1},\mathbf{p}^{k+1},\mathbf{w},\mathbf{y}^k,\mathbf{z}^k;\boldsymbol{\mu}^k,\boldsymbol{\xi}^k,\boldsymbol{\eta}^k\right),\\
\mathbf{y}^{k+1}&=\underset{\mathbf{y}}{\operatorname{argmin}}\,\mathscr{L}_{\mathscr{A}}\left(\mathbf{u}^{k+1},\mathbf{p}^{k+1},\mathbf{w}^{k+1},\mathbf{y},\mathbf{z}^k;\boldsymbol{\mu}^k,\boldsymbol{\xi}^k,\boldsymbol{\eta}^k\right),\\
\mathbf{z}^{k+1}&=\underset{\mathbf{z}}{\operatorname{argmin}}\,\mathscr{L}_{\mathscr{A}}\left(\mathbf{u}^{k+1},\mathbf{p}^{k+1},\mathbf{w}^{k+1},\mathbf{y}^{k+1},\mathbf{z};\boldsymbol{\mu}^k,\boldsymbol{\xi}^k,\boldsymbol{\eta}^k\right),\\
\boldsymbol{\mu}^{k+1}&=\boldsymbol{\mu}^k-\beta_1\left(\mathbf{w}^{k+1}-\mathbf{u}^{k+1}\right),\\
\boldsymbol{\xi}^{k+1}&=\boldsymbol{\xi}^k-\beta_2\left(\mathbf{y}^{k+1}-\varepsilon\left(\mathbf{u}^{k+1}\right)\right),\\
\boldsymbol{\eta}^{k+1}&=\boldsymbol{\eta}^k-\beta_3\left(\mathbf{z}^{k+1}-\nabla\mathbf{u}^{k+1}+\mathbf{p}^{k+1}\right).
\end{aligned} \tag{20}$$

3.2. Solution of the Subproblems. With the auxiliary $\mathbf{w}$, the $\mathbf{u}$ subproblem becomes quadratic and irrelevant to the constraint of the feasible set. It allows the following objective:

$$\mathbf{u}^{k+1}=\underset{\mathbf{u}}{\operatorname{argmin}}\frac{\beta_1}{2}\left\|\mathbf{w}^k-\mathbf{u}-\frac{\boldsymbol{\mu}^k}{\beta_1}\right\|_2^2+\frac{\beta_3}{2}\left\|\mathbf{z}^k-\nabla\mathbf{u}+\mathbf{p}^k-\frac{\boldsymbol{\eta}^k}{\beta_3}\right\|_2^2. \tag{21}$$

The minimization problem (21) can be solved by the following equation:

$$\mathbf{u}^{k+1}=\left(\frac{\beta_1}{\beta_3}\mathbf{I}+\nabla^T\nabla\right)^{-1}\left(\frac{\beta_1}{\beta_3}\left(\mathbf{w}^k-\frac{\boldsymbol{\mu}^k}{\beta_1}\right)+\nabla^T\left(\mathbf{z}^k+\mathbf{p}^k-\frac{\boldsymbol{\eta}^k}{\beta_3}\right)\right). \tag{22}$$

With the circulant boundary condition of images, we can solve (22) with several FFTs and IFFTs [8, 10].

Following the same way, the subproblem with respect to $\mathbf{p}$ is also quadratic and we have the objective functional as follows:

$$\mathbf{p}^{k+1}=\underset{\mathbf{p}}{\operatorname{argmin}}\frac{\beta_2}{2}\left\|\mathbf{y}^k-\varepsilon(\mathbf{p})-\frac{\boldsymbol{\xi}^k}{\beta_2}\right\|_2^2+\frac{\beta_3}{2}\left\|\mathbf{z}^k-\nabla\mathbf{u}^{k+1}+\mathbf{p}-\frac{\boldsymbol{\eta}^k}{\beta_3}\right\|_2^2. \tag{23}$$

Then, for $\mathbf{p}_1^{k+1}$, we have

$$\begin{aligned}
&\left(\beta_2\nabla_{x_1}^T\nabla_{x_1}+\frac{\beta_2}{2}\nabla_{x_2}^T\nabla_{x_2}+\beta_3\mathbf{I}\right)\mathbf{p}_1^{k+1}\\
&\quad=\beta_2\left[\nabla_{x_1}^T\left(\mathbf{y}_1^k-\frac{\boldsymbol{\xi}_1^k}{\beta_2}\right)+\nabla_{x_2}^T\left(\mathbf{y}_3^k-\frac{\boldsymbol{\xi}_3^k}{\beta_2}-\frac{1}{2}\nabla_{x_1}\mathbf{p}_2^k\right)\right]\\
&\qquad+\beta_3\left(\nabla_{x_1}\mathbf{u}^{k+1}+\frac{\boldsymbol{\eta}_1^k}{\beta_3}-\mathbf{z}_1^k\right),
\end{aligned} \tag{24}$$

and for $\mathbf{p}_2^{k+1}$, we have

$$\begin{aligned}
&\left(\beta_2\nabla_{x_2}^T\nabla_{x_2}+\frac{\beta_2}{2}\nabla_{x_1}^T\nabla_{x_1}+\beta_3\mathbf{I}\right)\mathbf{p}_2^{k+1}\\
&\quad=\beta_2\left[\nabla_{x_2}^T\left(\mathbf{y}_2^k-\frac{\boldsymbol{\xi}_2^k}{\beta_2}\right)+\nabla_{x_1}^T\left(\mathbf{y}_3^k-\frac{\boldsymbol{\xi}_3^k}{\beta_2}-\frac{1}{2}\nabla_{x_2}\mathbf{p}_1^{k+1}\right)\right]\\
&\qquad+\beta_3\left(\nabla_{x_2}\mathbf{u}^{k+1}+\frac{\boldsymbol{\eta}_2^k}{\beta_3}-\mathbf{z}_2^k\right),
\end{aligned} \tag{25}$$

where $\mathbf{p}_1$ and $\mathbf{p}_2$ are the combinations of $p_{i,j,1}$ and $p_{i,j,2}$ ($0\le i\le m$, $0\le j\le n$), respectively. Similar to the solution of (22), problems (24) and (25) can also be solved conveniently through several FFTs and IFFTs under the assumption of the circulant boundary condition.

The subproblem for $\mathbf{y}$ can be written as

$$\mathbf{y}^{k+1}=\underset{\mathbf{y}}{\operatorname{argmin}}\|\mathbf{y}\|_1+\frac{\beta_2}{2\alpha_0}\left\|\mathbf{y}-\varepsilon\left(\mathbf{p}^{k+1}\right)-\frac{\boldsymbol{\xi}^k}{\beta_2}\right\|_2^2. \tag{26}$$

Problem (26) can be solved component-wisely through the following 4-dimensional shrinkage operation:

$$\begin{aligned}
\mathbf{y}_{i,j}^{k+1}&=\max\left\{\left\|\varepsilon\left(\mathbf{p}^{k+1}\right)_{i,j}+\frac{\boldsymbol{\xi}_{i,j}^k}{\beta_2}\right\|_2-\frac{\alpha_0}{\beta_2},0\right\}\\
&\quad\times\frac{\varepsilon\left(\mathbf{p}^{k+1}\right)_{i,j}+\left(\boldsymbol{\xi}_{i,j}^k/\beta_2\right)}{\left\|\varepsilon\left(\mathbf{p}^{k+1}\right)_{i,j}+\left(\boldsymbol{\xi}_{i,j}^k/\beta_2\right)\right\|_2}\\
&\quad0\le i\le m,\quad 0\le j\le n.
\end{aligned} \tag{27}$$

The $\mathbf{z}$ subproblem is given by

$$\mathbf{z}^{k+1}=\underset{\mathbf{z}}{\operatorname{argmin}}\|\mathbf{z}\|_1+\frac{\beta_3}{2\alpha_1}\left\|\mathbf{z}-\nabla\mathbf{u}^{k+1}+\mathbf{p}^{k+1}-\frac{\boldsymbol{\eta}^k}{\beta_3}\right\|_2^2, \tag{28}$$

Input: $\mathbf{u}_0, c$.
(1) Initialize $\mathbf{u}^0, \mathbf{p}^0, \mathbf{w}^0, \mathbf{y}^0, \mathbf{z}^0, \boldsymbol{\mu}^0, \boldsymbol{\xi}^0, \boldsymbol{\eta}^0$. Set $k = 0$ and $\beta_1, \beta_2, \beta_3 > 0$.
(2) **while** stopping criterion is not satisfied, **do**
(3) Compute $\mathbf{u}^{k+1}$ according to (22);
(4) Compute $\mathbf{p}^{k+1}$ according to (24) and (25);
(5) Compute $\mathbf{y}^{k+1}$ according to (27);
(6) Compute $\mathbf{z}^{k+1}$ according to (29);
(5) **if** (32) holds, **then**
(6) $\lambda^{k+1} = 0$ and $\mathbf{w}^{k+1} = \mathbf{u}^{k+1} + (\boldsymbol{\mu}^k/\beta_1)$;
(7) **else**
(8) Update λ^{k+1} and $\mathbf{w}^{k+1}$ according to (34) and (31);
(9) **end if**
(10) Update $\boldsymbol{\mu}^{k+1}, \boldsymbol{\xi}^{k+1}$, and $\boldsymbol{\eta}^{k+1}$ according to (20);
(11) $k = k + 1$;
(12) **end while**
(13) **return** $\mathbf{u}^{k+1}$.

Algorithm 1: **Algorithm TGV^{2}ID-ADMM:** Second-order TGV-regularized image denoising with ADMM.

and it can be solved component-wisely through the following 2-dimensional shrinkage operation:

$$\mathbf{z}_{i,j}^{k+1} = \max\left\{\left\|\left(\nabla\mathbf{u}^{k+1}\right)_{i,j} + \frac{\boldsymbol{\eta}_{i,j}^k}{\beta_3} - \mathbf{p}_{i,j}^{k+1}\right\|_2 - \frac{\alpha_1}{\beta_3}, 0\right\} \times \frac{\left(\nabla\mathbf{u}^{k+1}\right)_{i,j} + \left(\boldsymbol{\eta}_{i,j}^k/\beta_3\right) - \mathbf{p}_{i,j}^{k+1}}{\left\|\left(\nabla\mathbf{u}^{k+1}\right)_{i,j} + \left(\boldsymbol{\eta}_{i,j}^k/\beta_3\right) - \mathbf{p}_{i,j}^{k+1}\right\|_2}, \quad 0 \le i \le m,\ 0 \le j \le n. \tag{29}$$

The subproblem with respect to $\mathbf{w}$ can be written as

$$\begin{aligned}\mathbf{w}^{k+1} &= \operatorname*{argmin}_{\mathbf{w}} I_\Phi(\mathbf{w}) + \frac{\beta_1}{2}\left\|\mathbf{w} - \left(\mathbf{u}^{k+1} + \frac{\boldsymbol{\mu}^k}{\beta_1}\right)\right\|_2^2 \\ &= \operatorname*{argmin}_{\mathbf{w}} \frac{\lambda^{k+1}}{2}\|\mathbf{w} - \mathbf{u}_0\|_2^2 + \frac{\beta_1}{2}\left\|\mathbf{w} - \left(\mathbf{u}^{k+1} + \frac{\boldsymbol{\mu}^k}{\beta_1}\right)\right\|_2^2.\end{aligned} \tag{30}$$

Consequently, the solution of problem (30) is

$$\mathbf{w}^{k+1} = \frac{\lambda^{k+1}\mathbf{u}_0 + \beta_1\left(\mathbf{u}^{k+1} + \left(\boldsymbol{\mu}^k/\beta_1\right)\right)}{\lambda^{k+1} + \beta_1}. \tag{31}$$

The solution of λ^{k+1} falls into two cases according to the range of $\mathbf{u}^{k+1} + (\boldsymbol{\mu}^k/\beta_1)$. On one hand, if

$$\left\|\mathbf{u}^{k+1} + \frac{\boldsymbol{\mu}^k}{\beta_1} - \mathbf{u}_0\right\|_2^2 \le c, \tag{32}$$

we can set $\lambda^{k+1} = 0$, and, obviously, $\mathbf{w}^{k+1} = \mathbf{u}^{k+1} + \boldsymbol{\mu}^k/\beta_1$ satisfies the feasible set constraint. On the other hand, if (32) is not true, $\mathbf{w}^{k+1}$ should fulfill the following equation:

$$\left\|\mathbf{w}^{k+1} - \mathbf{u}_0\right\|_2^2 = c. \tag{33}$$

Substituting (31) into (33), we get

$$\lambda^{k+1} = \frac{\beta_1\left\|\mathbf{u}^{k+1} + \left(\boldsymbol{\mu}^k/\beta_1\right) - \mathbf{u}_0\right\|_2}{\sqrt{c}} - \beta_1. \tag{34}$$

The resulting image denoising algorithm is summarized in Algorithm 1 TGV^{2}ID-ADMM.

The adoption of the variable $\mathbf{w}$ is essential for the adaptive estimate of the regularization parameter λ. With the assistance of $\mathbf{w}$, $\mathbf{u}$ is liberated out from the constraint of the feasible set. Thus, the update of λ is free from the disturbance of the update of $\mathbf{u}$, and a closed form for updating λ is achieved in each step without inner iteration. From functionals (16) and (18) we learn that, by setting $(\alpha_0, \alpha_1) = (0, 1)$, $(\mathbf{y}, \mathbf{p}) = (\mathbf{0}, \mathbf{0})$, and $\beta_2 = 0$, Algorithm TGV^{2}ID-ADMM will degenerate to a TV-based denoising algorithm, and we denote this case as TGV^{1}ID-ADMM.

The convergence of Algorithm TGV^{2}ID-ADMM follows from the convergence analysis for the TV-based ADMM in [11, 25], due to the convex property of TGV$^2_{\boldsymbol{\alpha}}$. In this paper, we do not repeat the lengthy analysis procedure. However, we have the following essential convergence theorem for the proposed method.

Theorem 1. *For fixed $\beta_1, \beta_2, \beta_3 > 0$, the sequence $\{\mathbf{u}^k, \mathbf{p}^k, \mathbf{w}^k, \mathbf{y}^k, \mathbf{z}^k, \boldsymbol{\mu}^k, \boldsymbol{\xi}^k, \boldsymbol{\eta}^k, \lambda^k\}$ generated by Algorithm TGV^{2}ID-ADMM from any initial point $(\mathbf{u}^0, \mathbf{p}^0, \mathbf{w}^0, \mathbf{y}^0, \mathbf{z}^0, \boldsymbol{\mu}^0, \boldsymbol{\xi}^0, \boldsymbol{\eta}^0)$ converges to $(\mathbf{u}^*, \mathbf{p}^*, \mathbf{w}^*, \mathbf{y}^*, \mathbf{z}^*, \boldsymbol{\mu}^*, \boldsymbol{\xi}^*, \boldsymbol{\eta}^*, \lambda^*)$, where $\mathbf{u}^*$ is the solution of functional (15) and λ^* is the regularization parameter corresponding to the feasible set constraint $\mathbf{u} \in \Phi$.*

4. Experiment Results

In this section, we illustrate the effectiveness of the proposed algorithm on suppressing staircasing effect and removing Gaussian noise in image. Besides, we also show the robustness

Table 1: Results of the staircasing effect reduction experiment.

Piecewise affine image denoising with $\sigma = 15$				
Model	RMSE	PSNR	Iterations	Time (s)
Noised	14.99	24.61	—	—
TGV^2	1.89	42.61	155	18.21
TGV^1	2.40	39.81	128	7.76

of the results with respect to the penalty parameters. We performed our algorithm under MATLAB v7.8.0 and Windows 7 on a PC with an Intel Core (TM) i5 CPU at 3.20 GHz and 8 GB of RAM.

The root mean squared error (RMSE) and the peak signal-to-noise ratio (PSNR) used in comparison are defined as

$$\begin{aligned} \text{RMSE} &= \frac{\left\|\mathbf{u} - \mathbf{u}_{\text{clean}}\right\|_2}{\sqrt{mn}}, \\ \text{PSNR} &= 10\log_{10}\left(\frac{255^2 \cdot mn}{\left\|\mathbf{u} - \mathbf{u}_{\text{clean}}\right\|_2^2}\right), \end{aligned} \tag{35}$$

where $\mathbf{u}_{\text{clean}}$ is the original image that contains no noise. Besides, in subsections 4.1 and 4.2, we set the penalty parameters as $\beta_1 = 10^{(\text{BSNR}/10-1)} \times \beta$ and $\beta_2 = \beta_3 = \beta = 0.3$ for TGV^2ID-ADMM ($\beta_2 = 0$ for TGV^1ID-ADMM) to achieve consistently promising result with fast speed, where BSNR is the blurred signal-to-noise ratio defined by $\text{BSNR} = 10\log_{10}(\text{var}(\mathbf{u}_0)/\sigma^2)$ ($\text{var}(\mathbf{u}_0)$ denotes the variance of $\mathbf{u}_0$).

4.1. Staircasing Effect Reduction by the Proposed Method. We first compare Algorithm TGV^2ID-ADMM with Algorithm TGV^1ID-ADMM to illustrate the effectiveness of $TGV^2_{\boldsymbol{\alpha}}$ model in staircasing effect reduction. We use $\|\mathbf{u}^{k+1} - \mathbf{u}^k\|_2^2 / \|\mathbf{u}^k\|_2^2 \leq 10^{-6}$ as the stopping criteria for these two algorithms, where $\mathbf{u}^k$ denotes the restored result in the kth iteration. For the second-order case, we set $(\alpha_0, \alpha_1) = (3, 1)$, whereas for the one-order case, we set $(\alpha_0, \alpha_1) = (0, 1)$.

In this experiment, we use a synthetic piecewise affine image shown in Figure 1 as the test image. The original image is contaminated by Gaussian noise of standard variance $\sigma = 15$ at first. Then we imply TGV^2ID-ADMM and TGV^1ID-ADMM to remove the noise. Table 1 shows the results in terms of RMSE, PSNR, total iterations, and CPU time. The ground truth, noised, and restored images by the two algorithms are displayed in Figure 1. Furthermore, for better visualization, we additionally provide the three-dimensional close-ups of the marked regions of the two restored images in Figure 1. From Table 1 we observe that TGV^2ID-ADMM does better than TGV^1ID-ADMM in terms of both RMSE and PSNR. Figure 1 shows that the denoised image of TGV^2ID-ADMM almost contains no artificial edges in affine regions. In contrast, the restored result of TGV^1ID-ADMM contains obvious staircasing effect in affine regions. The three-dimensional closed-ups vividly demonstrate this phenomenon. This illustrates that our TGV-based algorithm is effective in staircasing effect reduction.

Table 1 also shows that, to accomplish the denoising task, TGV^2ID-ADMM usually costs more CPU time than TGV^1ID-ADMM, since $TGV^2_{\boldsymbol{\alpha}}$ model involves much more calculation. However, the cost is worthy due to the impressive improvement on both quantitative and qualitative restoration quality. Figure 2 displays the evolutions of λs and PSNRs achieved by the two algorithms. It is learnt that, the regularization parameters of both converge to the optimal points at last, which guarantees the automatic implementation of the two algorithms.

4.2. Comparison in Accuracy. In this subsection, we compare TGV^2ID-ADMM with the other two famous adaptive TV-based denoising algorithms: Chambolle's projection algorithm [23] and Split Bregman algorithm [24], both possessing public online implementations at "http://www.ipol.im/". Two natural images, Lena and Peppers both of size 512×512 shown in Figure 3, are used for comparison. The parameter setting for TGV^2ID-ADMM is the same as that in the previous subsection. We obtain the test results of the two competitors through online experimental operation.

We add Gaussian noise of standard variances of 20, 30, and 40 to Lena and Peppers to obtain the noised observations, respectively. Then we apply these three algorithms to restore the noisy images. Table 2 shows the comparison results in terms of RMSE and PSNR. The best result for each comparison item is highlighted in bold type font. Table 2 shows that TGV^2ID-ADMM holds superiority on both RMSE and PSNR for all the tested cases. Figure 4 displays the noised Lena under Gaussian noise of $\sigma = 30$ and the restorations by the three algorithms, whereas Figure 5 exhibits the noised Peppers under Gaussian noise of $\sigma = 40$ and the corresponding restorations. Figures 4 and 5 demonstrate that TGV^2ID-ADMM obtains results with better visual impression and efficiently suppresses the staircasing effect. In contrast, both TV-based Chambolle's projection algorithm and TV-based Split Bregman algorithm achieve results with obvious staircasing effect. Since we apply test images with different levels of noise, the robustness of our algorithm towards the noise level is verified to a certain extent.

4.3. Solution Robustness with Respect to the Penalty Parameters. Although the positive assumption of penalty parameters is sufficient for the convergence of ADMM, the results of

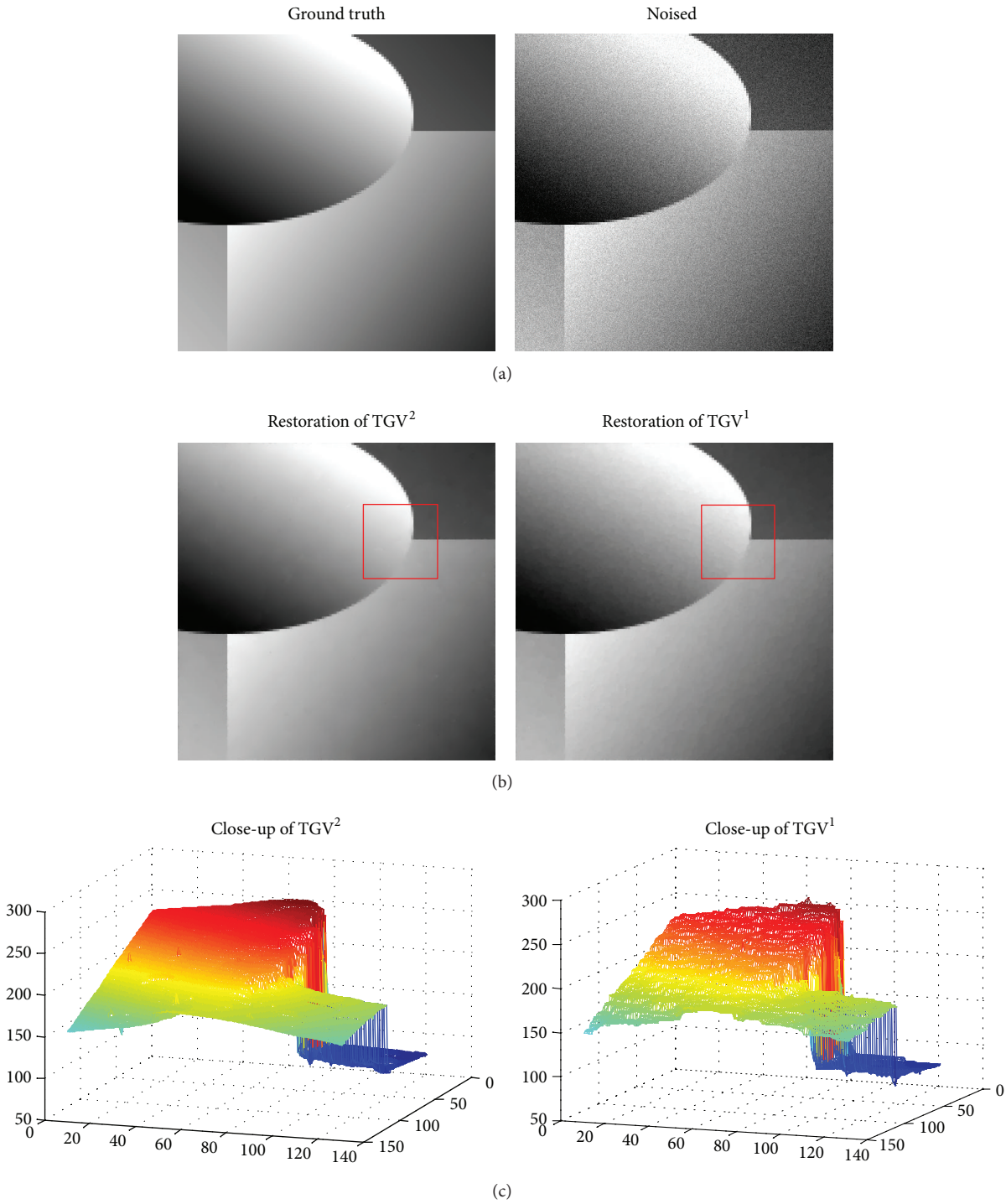
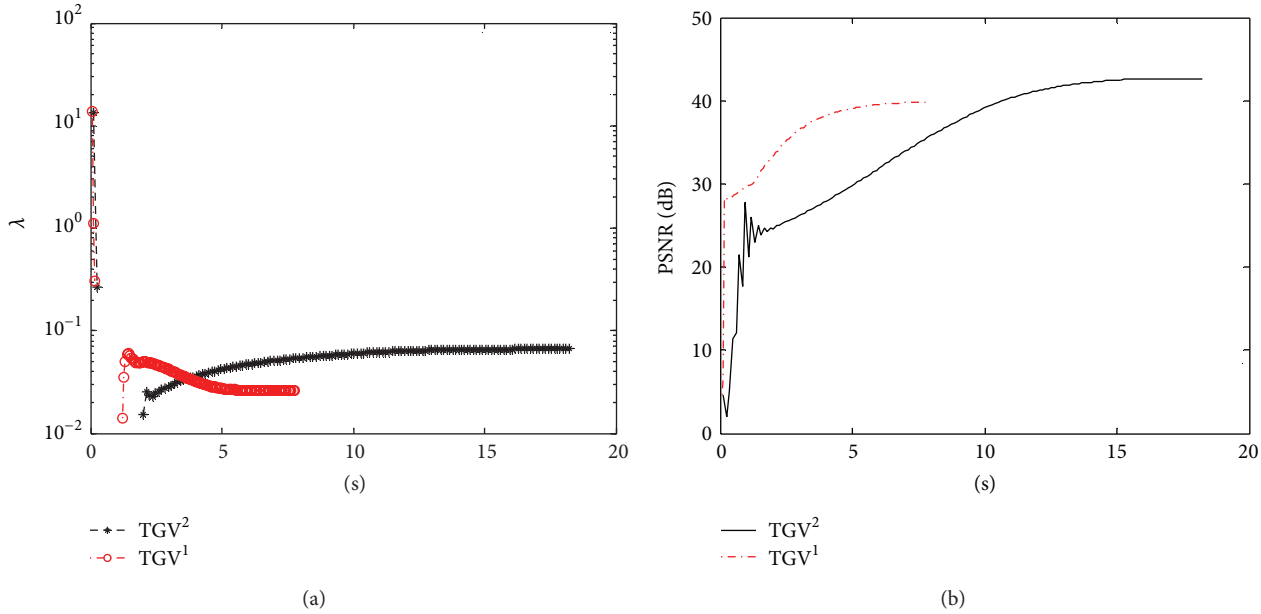


FIGURE 1: First row: ground truth and Gaussian noised ($\sigma = 15$) piecewise affine images; second row: restored images by TGV^{2}ID-ADMM and TGV^{1}ID-ADMM; third row: three-dimensional close-ups of the marked regions in the two restored images.

ADMM are commonly influenced by the choice of the penalty parameters to a certain extent in practice. As suggested by a referee, we add an experiment to show the robustness of the results of TGV^{2}ID-ADMM with respect to the penalty parameters, under the two denoising background problems mentioned above, that is, the Lena denoising problem under Gaussian noise of $\sigma = 30$ and the Peppers denoising problem under Gaussian noise of $\sigma = 40$. We still set $\beta_1 = 10^{(\mathrm{BSNR}/10-1)} \times \beta$ and $\beta_2 = \beta_3 = \beta$ but change β from 0.01 to 1 with a step size of 0.01. In Figure 6, we plot PSNR versus β for the denoised Lena and Peppers. Figure 6 demonstrates that the optimal β should be focalized in $[0.2, 0.3]$ and its

Table 2: Comparison results in accuracy.

σ	Noised		TGV^{2}ID-ADMM		Chambolle		Split Bregman	
	RMSE	PSNR	RMSE	PSNR	RMSE	PSNR	RMSE	PSNR
				Lena 512 × 512				
20	19.99	22.11	**7.09**	**31.12**	7.40	30.75	8.05	30.01
30	30.00	18.59	**8.67**	**29.37**	9.45	28.62	9.95	28.17
40	39.97	16.10	**10.06**	**28.07**	10.54	27.67	11.66	26.80
				Peppers 512 × 512				
20	19.96	22.13	**7.05**	**31.17**	7.38	30.77	7.82	30.27
30	29.98	18.59	**8.68**	**29.36**	9.43	28.64	9.92	28.20
40	40.10	16.07	**9.74**	**28.36**	10.74	27.51	12.38	26.28



(a) (b)

Figure 2: Evolutions of λs (a) and PSNRs (b) achieved by TGV^{2}ID-ADMM (TGV^2) and TGV^{1}ID-ADMM (TGV^1).

(a) (b)

Figure 3: Test images: Lena and Peppers.

Figure 4: Noised Lena under Gaussian noise of $\sigma = 30$ and the restorations by TGV^{2}ID-ADMM, Chambolle, and Split Bregman, respectively.

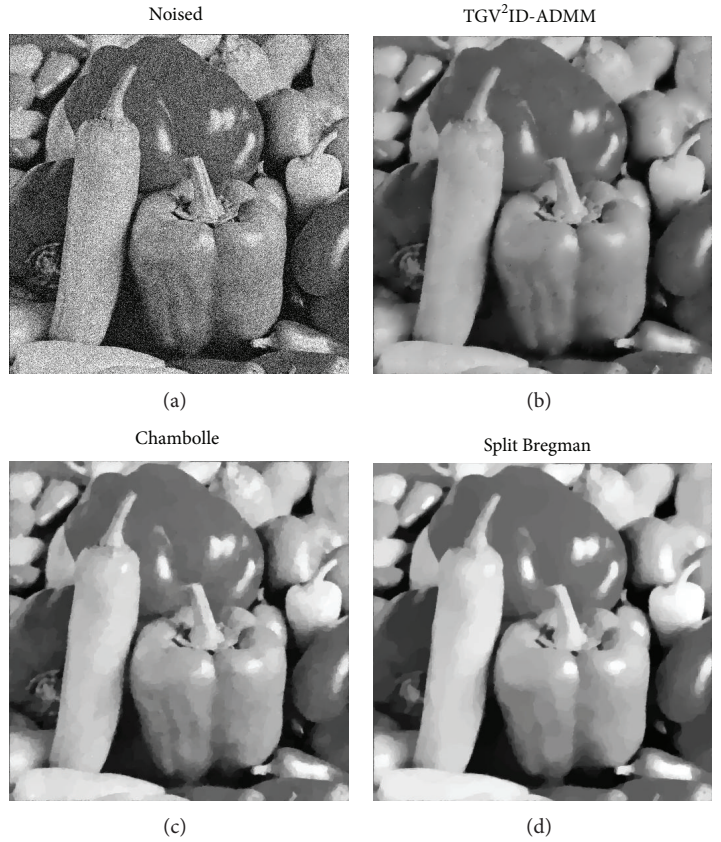
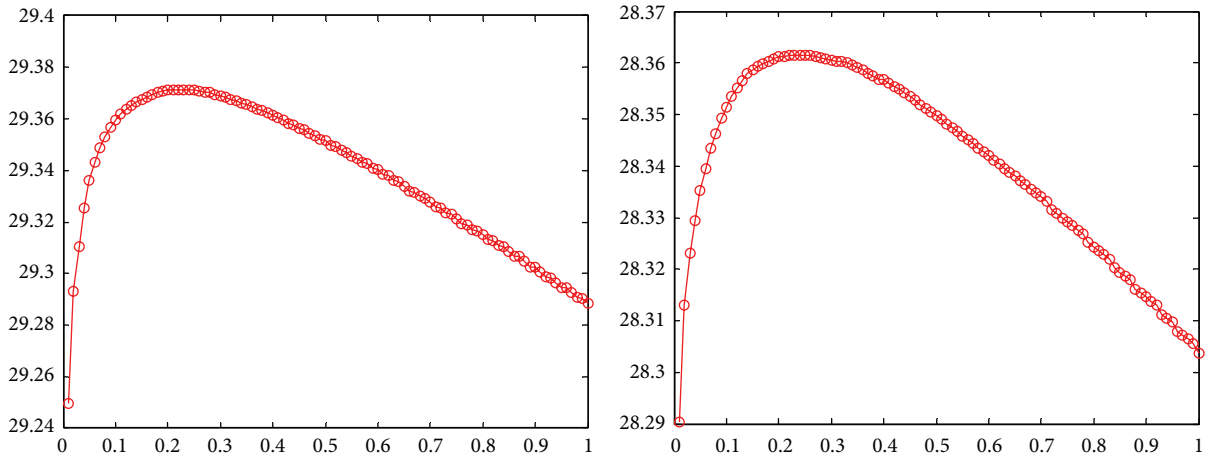


Figure 5: Noised Peppers under Gaussian noise of $\sigma = 40$ and the restorations by TGV^{2}ID-ADMM, Chambolle, and Split Bregman, respectively.



(a) Lena image under Gaussian noise of $\sigma = 30$

(b) Peppers image under Gaussian noise of $\sigma = 40$

FIGURE 6: PSNR versus β for (a) Lena image under Gaussian noise of $\sigma = 30$ and (b) Peppers image under Gaussian noise of $\sigma = 40$, respectively. The absolute error between the maximum and the minimum of PSNR for each image is less than 0.18 dB.

location is robust towards the variation of image and noise level. The results of our method possess sufficient robustness with respect to the variation of penalty parameters to a certain extent, since the absolute error between the maximum and the minimum of PSNR is less than 0.18 dB in the experiment, and this error could not introduce obvious distinction in visual quality. In the former two experiments, the setting of $\beta = 0.3$ is approximately optimal for the proposed algorithm.

5. Concluding Remarks

We propose an adaptive TGV-based model for noise removal in this paper. The variable splitting (VS) and the classical augmented Lagrangian method are used to handle the proposed model. From the experimental results, we observe that the proposed algorithm is effective in suppressing staircasing effect and preserving edges in images, and it is superior to some other famed adaptive denoising methods both in quantitative and in qualitative assessment. Besides, our work can be smoothly generalized to image deblurring problems.

Appendix

The Equivalent Definition of $\mathrm{TGV}_{\alpha}^{2}$

In discrete version, we have

$$\mathrm{TGV}_{\boldsymbol{\alpha}}^{2}(\mathbf{u}) = \max_{\mathbf{v},\mathbf{d}} \left\{ \langle \mathbf{u}, \operatorname{div} \mathbf{d} \rangle \mid \operatorname{div} \mathbf{v} = \mathbf{d}, \; \|\mathbf{v}\|_{\infty} \le \alpha_0, \|\mathbf{d}\|_{\infty} \le \alpha_1 \right\}, \tag{A.1}$$

where

$$\|\mathbf{v}\|_{\infty} = \max_{i,j} \left(v_{i,j,1}^2 + v_{i,j,2}^2 + 2v_{i,j,3}^2 \right)^{1/2},$$

$$\|\mathbf{d}\|_{\infty} = \max_{i,j} \left(d_{i,j,1}^2 + d_{i,j,2}^2 \right)^{1/2}$$

$$\left(\mathbf{v}_{i,j} = \begin{bmatrix} v_{i,j,1} & v_{i,j,3} \\ v_{i,j,3} & v_{i,j,2} \end{bmatrix}, \mathbf{d}_{i,j} = \begin{bmatrix} d_{i,j,1}, d_{i,j,2} \end{bmatrix} \right). \tag{A.2}$$

Therefore, according to the Lagrange duality, we have

$$\begin{aligned} \mathrm{TGV}_{\boldsymbol{\alpha}}^{2}(\mathbf{u}) &= \min_{\mathbf{p}} \max_{\|\mathbf{v}\|_{\infty} \le \alpha_0, \|\mathbf{d}\|_{\infty} \le \alpha_1} \langle \mathbf{u}, \operatorname{div} \mathbf{d} \rangle + \langle \mathbf{p}, \mathbf{d} - \operatorname{div} \mathbf{v} \rangle \\ &= \min_{\mathbf{p}} \max_{\|\mathbf{v}\|_{\infty} \le \alpha_0, \|\mathbf{d}\|_{\infty} \le \alpha_1} \langle -\nabla \mathbf{u}, \mathbf{d} \rangle + \langle \mathbf{p}, \mathbf{d} \rangle + \langle \varepsilon(\mathbf{p}), \mathbf{v} \rangle \\ &= \min_{\mathbf{p}} \max_{\|\mathbf{v}\|_{\infty} \le \alpha_0, \|\mathbf{d}\|_{\infty} \le \alpha_1} \langle \mathbf{p} - \nabla \mathbf{u}, \mathbf{d} \rangle + \langle \varepsilon(\mathbf{p}), \mathbf{v} \rangle \\ &= \min_{\mathbf{p}} \alpha_0 \|\varepsilon(\mathbf{p})\|_1 + \alpha_1 \|\nabla \mathbf{u} - \mathbf{p}\|_1. \end{aligned} \tag{A.3}$$

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 Editor Fatih Yaman and anonymous referees for their valuable comments. Their help has greatly enhanced the quality of this paper. This work was partially supported by the National Natural Science Foundation of China under Grant nos. 61203189, 61104223, and

61374120 and the National Science Fund for Distinguished Young Scholars of China under Grant no. 61025014.

References

[1] L. I. Rudin, S. Osher, and E. Fatemi, "Nonlinear total variation based noise removal algorithms," *Physica D: Nonlinear Phenomena*, vol. 60, no. 1–4, pp. 259–268, 1992.

[2] T. Chan, S. Esedoglu, F. Park, and A. Yip, "Recent developments in total variation image restoration," in *Mathematical Models of Computer Vision*, Springer, New York, NY, USA, 2005.

[3] Y. L. You and M. Kaveh, "Fourth-order partial differential equations for noise removal," *IEEE Transactions on Image Processing*, vol. 9, no. 10, pp. 1723–1730, 2000.

[4] M. Lysaker, A. Lundervold, and X.-C. Tai, "Noise removal using fourth-order partial differential equation with applications to medical magnetic resonance images in space and time," *IEEE Transactions on Image Processing*, vol. 12, no. 12, pp. 1579–1589, 2003.

[5] T. F. Chan, S. Esedoglu, and F. Park, "A fourth order dual method for staircase reduction in texture extraction and image restoration problems," in *Proceedings of the 17th IEEE International Conference on Image Processing (ICIP '10)*, pp. 4137–4140, Los Angeles, Calif, USA, September 2010.

[6] M. R. Hajiaboli, "An anisotropic fourth-order diffusion filter for image noise removal," *International Journal of Computer Vision*, vol. 92, no. 2, pp. 177–191, 2011.

[7] T. Liu and Z. Xiang, "Image restoration combining the second-order and fourth-order PDEs," *Mathematical Problems in Engineering*, vol. 2013, Article ID 743891, 7 pages, 2013.

[8] Y. Wang, J. Yang, W. Yin, and Y. Zhang, "A new alternating minimization algorithm for total variation image reconstruction," *SIAM Journal on Imaging Sciences*, vol. 1, no. 3, pp. 248–272, 2008.

[9] N. B. Brás, J. Bioucas-Dias, R. C. Martins, and A. C. Serra, "An alternating direction algorithm for total variation reconstruction of distributed parameters," *IEEE Transactions on Image Processing*, vol. 21, no. 6, pp. 3004–3016, 2012.

[10] C. He, C. Hu, W. Zhang, B. Shi, and X. Hu, "Fast total-variation image deconvolution with adaptive parameter estimation via split Bregman method," *Mathematical Problems in Engineering*, vol. 2014, Article ID 617026, 9 pages, 2014.

[11] T. Goldstein, B. O'Donoghue, S. Setzer, and R. Baraniuk, "Fast alternating direction optimization methods," UCLA CAM Report, 2012.

[12] T. Chan, A. Marquina, and P. Mulet, "High-order total variation-based image restoration," *SIAM Journal on Scientific Computing*, vol. 22, no. 2, pp. 503–516, 2001.

[13] O. Scherzer, "Denoising with higher order derivatives of bounded variation and an application to parameter estimation," *Computing*, vol. 60, no. 1, pp. 1–27, 1998.

[14] A. Chambolle and P.-L. Lions, "Image recovery via total variation minimization and related problems," *Numerische Mathematik*, vol. 76, no. 2, pp. 167–188, 1997.

[15] G. Dal Maso, I. Fonseca, G. Leoni, and M. Morini, "A higher order model for image restoration: the one-dimensional case," *SIAM Journal on Mathematical Analysis*, vol. 40, pp. 2351–2391, 2009.

[16] B. Shi, Z. F. Pang, and Y. F. Yang, "Image restoration based on the hybrid total-variation-type model," *Abstract and Applied Analysis*, vol. 2012, Article ID 376802, 30 pages, 2012.

[17] K. Bredies, K. Kunisch, and T. Pock, "Total generalized variation," *SIAM Journal on Imaging Sciences*, vol. 3, no. 3, pp. 492–526, 2010.

[18] Yu. Nesterov, "A method for solving a convex programming problem with convergence rate O(1/k2)," *Soviet Mathematics Doklady*, vol. 27, pp. 372–376, 1983.

[19] R. Glowinski and P. Le Tallec, *Augmented Lagrangians and Operator-Splitting Methods in Nonlinear Mechanics*, Studies in Applied Mathematics 9, SIAM, Philadelphia, Pa, USA, 1989.

[20] S. Xie and S. Rahardja, "Alternating direction method for balanced image restoration," *IEEE Transactions on Image Processing*, vol. 21, no. 11, pp. 4557–4567, 2012.

[21] W. Deng and W. Yin, "On the global and linear convergence of the generalized alternating direction method of multipliers," UCLA CAM Report Cam 12-52, 2012.

[22] W. Guo, J. Qin, and W. Yin, "A new detail-preserving regularity scheme," UCLA CAM Report cam13-04, 2013.

[23] J. Duran, B. Coll, and C. Sbert, "Chambolle's pojection algorithm for total variation denoising," *Image Processing on Line*, vol. 3, pp. 311–331, 2013.

[24] P. Getreuer, "Rudin-Osher-Fatemi total variation denoising using split Bregman," *Image Processing on Line*, vol. 2, pp. 74–95, 2012.

[25] C. Wu and X.-C. Tai, "Augmented lagrangian method, dual methods, and split Bregman iteration for ROF, vectorial TV, and high order models," *SIAM Journal on Imaging Sciences*, vol. 3, no. 3, pp. 300–339, 2010.

Hindawi Publishing Corporation
Mathematical Problems in Engineering
Volume 2014, Article ID 653208, 12 pages
http://dx.doi.org/10.1155/2014/653208

Research Article

Sparse Scenario Imaging for Active Radar in the Forward-Looking Direction

Jun Wang, Fenggang Yan, Yinan Zhao, and Xiaolin Qiao

School of Electronics and Information Engineering, Harbin Institute of Technology, Harbin, Heilongjiang 150001, China

Correspondence should be addressed to Jun Wang; johnwangstudio@gmail.com

Received 8 April 2014; Accepted 29 May 2014; Published 3 July 2014

Academic Editor: Caner Özdemir

The resolution of multiple targets at the same range cell but different angles in the forward-looking direction is of great trouble for active radar. Based on compressive sensing (CS) framework, a sparse scenario imaging approach using joint angle-Doppler representation basis is proposed, which employs multisensor and single-receiver channel hardware architecture. Firstly, the joint angle-Doppler representation basis is formulated using the Doppler dictionary, and then the radar returns during multiple pulse repetition periods are modeled as the measurements with respect to a stationary sparse target scenario via the joint representation basis; in the end, the image of sparse target scenario is recovered using the single-receiver echoes. Numerical experiments demonstrate that the proposed method can provide an image of the spatial sparse scenario at the same range for active radar in the forward-looking direction.

1. Introduction

In the forward-looking direction, the resolution of multiple targets at the same range cell but different angles is of great trouble for active radar because traditional Doppler beam sharpening (DBS) and synthetic aperture radar (SAR) techniques can be employed to improve the cross-range resolution only in the squint and side-looking direction. Conventional multiple signal classification (MUSIC) and estimation of signal parameters via rotational invariance techniques (ESPRIT) algorithms which can obtain direction-of-arrival (DOA) of multiple targets can be difficultly employed in active radar because of the need of multireceiver structure and the decayed performance with coherent signals circumstance.

Compressive sensing (CS) [1, 2] is emerged in the past few years which states that a sparse signal can be economically acquired in a more economical way. A signal $\mathbf{x} \in \mathbb{C}^N$ which can be written in $\mathbf{x} = \mathbf{\Psi s}$ where $\mathbf{\Psi}$ is a representation basis is defined as a K-sparse signal if the number of nonzero elements of vector $\mathbf{s}$ is not larger than K. CS suggests that the sparse signal $\mathbf{x}$ can be reconstructed using $M \ll N$ measurements $\mathbf{y} = \mathbf{\Phi x} + \mathbf{n}$ if equivalent dictionary $\mathbf{D} = \mathbf{\Phi\Psi}$ satisfies RIP [1], where $\mathbf{\Phi} \in \mathbb{C}^{M\times N}$ is the measurement matrix and $\mathbf{n} \in \mathbb{C}^M$ is additive noise. The reconstructed $\hat{\mathbf{x}}$ can be obtained by solving a convex optimization problem as follows:

$$\min_{\hat{\mathbf{s}}\in\mathbb{C}^N} \|\hat{\mathbf{s}}\|_1, \quad \text{s.t. } \left\|\mathbf{y} - \mathbf{D}\hat{\mathbf{s}}\right\|_2 \le \varepsilon, \tag{1}$$

where $\varepsilon = \|\mathbf{n}\|_2$ and $\|\cdot\|_p$ denotes ℓp-norm.

DOA estimation problem generally faces a sparse target scenario in the spatial domain; consequently the application of CS in DOA estimation has been concerned recently. In [3, 4], DOA estimation is modeled as single measurement vector problem (SMV) and, in [5, 6], is solved using joint sparse reconstruction algorithm SOMP [7] based on multiple measurement vector model (MMV) [8]. Numerical experiments show that the DOA estimation algorithms based on CS exhibit better performance than traditional subspace algorithms such as MUSIC and ESPRIT, especially at lower signal-to-noise ratio (SNR) and with coherent signal circumstance. Inspired by Bayesian CS [9] and MT-CS [10] framework, [11–13] provide more accurate DOA estimation than traditional algorithms using appropriate priors with respect to hyperparameters. The compressive MUSIC [14] can approach

the optimal l_0-bound with finite number of snapshots even in cases where the signals are linear dependent. In [15], an SAR image compression approach based on CS shows better performance on preserving the important features than JPEG and JPEG2000, and the CS TomoSAR theory in [16] has superresolution properties and point localization accuracies.

Motivated by the better performance in DOA estimation and cross-range resolution in SAR based on CS framework, we develop a sparse scenario imaging strategy for active radar in the forward-looking direction which employs multisensor single-receiver channel structure.

2. Measurement Model of Sparse Scenario

In this section, the measurement model with respect to sparse target scenario is presented based on the joint angle-Doppler representation basis.

Traditional DOA estimation algorithms such as MUSIC and ESPRIT exploit relative phases between the outputs of multiple array sensors to determine multiple targets. In the proposed strategy, multiple equivalent sensors are formed in phased array radar (PAR) in order to provide a multisensor output during the reception. In nature, each equivalent sensor is a subarray which consists of multiple array elements. As shown in Figure 1, PAR forms M beams pointing to target scenario Ω using M subarrays during the reception and only one beam for illumination. The phase centers of M equivalent sensors can be expressed as $(x_m, y_m, 0),\ m = 0, \ldots, M-1$.

2.1. Measurement Model Using Multisensor. As depicted in Figure 2, the interest spatial area is restricted in elevation $\theta_{\max}$ and is discretized into N spatial grids $\Omega = \{\mathbf{u}_n, n = 0, \ldots, N-1\}$, where $\mathbf{u}_n = [u_n, v_n, w_n]^T$ is the cosine vector corresponding to nth grid which has elevation angle θ_n and azimuth angle φ_n.

Assuming that the radar is moving to Ω along with z axis as shown in Figure 2, let $\hat{t}$ denote fast time, and let $t_r = rT_r$ denote rth slow time where T_r is pulse repetition interval (PRI), and then the full time $t = t_r + \hat{t}$ is abbreviated to $(\hat{t}, t_r)$. The illumination waveform is $p(\hat{t}) = R(\hat{t})\tilde{p}(\hat{t})$, where $R(\hat{t}) = 1,\ 0 \le \hat{t} \le \tau;\ R(\hat{t}) = 0,\ \hat{t} > \tau$, is a window function and $\tilde{p}(\hat{t})$ is a wideband waveform. For simplicity, let $(\hat{t}, \mathbf{u}_n)$ denote the target position in nth spatial grid $\mathbf{u}_n$ at the range cell $\hat{t}$. Due to noncooperation property of target for active radar, the corresponding Doppler is not a prior. With respect to an arbitrary target on $(\hat{t}_T, \mathbf{u}_n)$, the returns of the reference array sensor whose coordinate is $(0, 0, 0)$ in one observation period including R PRIs can be expressed as follows:

$$\beta_n(\hat{t}_T)\, p(\hat{t} - \hat{t}_T) \exp\left[j2\pi f_n(\hat{t} + t_r)\right], \quad r = 0, \ldots, R-1, \tag{2}$$

where $\beta_n(\hat{t}_T)$ is the amplitude of target on $(\hat{t}_T, \mathbf{u}_n)$, and f_n is the corresponding unknown Doppler. The subscript "n" of $\beta_n(\hat{t}_T)$ and f_n denotes the corresponding term about the target from the nth spatial grid $\mathbf{u}_n \in \Omega$.

A sparse target scenario $E = \{(\hat{t}_T, \mathbf{u}_n), \mathbf{u}_n \in T\}$ where $K \ll N$ targets are located in $T \subset \Omega$ at the same range cell $\hat{t}_T$ is considered, where $T = \{\mathbf{u}_n, n \in I_T\}$ and I_T is the target index set of T in spatial domain Ω. The index set I_T satisfies the equation $|\operatorname{supp}(I_T)| = K$, where $\operatorname{supp}(\cdot)$ denotes the support operation and $|\cdot|$ denotes the element number of set. It is of great trouble for traditional active radar to determine such a target scenario in the forward-looking direction.

In order to express the radar returns from target scenario E, the Doppler dictionary $F = \{f_s^d\}_{s=0}^{S-1}$ including $S \ge K$ frequencies is employed here to settle the unknown Doppler problem in (2). The Doppler dictionary F can be seen as the estimation about the unknown target Doppler $F_D = \{f_n, n \in I_T\}$ of scenario E, and the estimated method will be given in Section 3. Due to the consideration on estimation error, we assume that there are more frequencies in F than actual Doppler assemble F_D; that is, $S \ge K$. Based on the utilization of F, we only "know" the estimated Doppler frequencies about scenario E, but we "do not know" the relationship between the frequencies $f_s^d,\ s = 0, \ldots, S-1$ in F and target directions $\mathbf{u}_n,\ n \in I_T$ in T. Therefore, an amplitude vector $\boldsymbol{\beta}_n(\hat{t}_T) = [\beta_{n,0}(\hat{t}_T), \ldots, \beta_{n,S-1}(\hat{t}_T)]^T$ corresponding to all S frequencies in F is defined for the target located in arbitrary nth grid $\mathbf{u}_n$, where the sth element of $\beta_{n,s}(\hat{t}_T),\ s = 0, \ldots, S-1$, denotes the amplitude when the Doppler of the target in $\mathbf{u}_n$ is the sth frequency $f_s^d \in F,\ s = 0, \ldots, S-1$. Assuming that the actual Doppler f_n of the target in $\mathbf{u}_n$ is equal to the ith frequency $f_i^d \in F$, the corresponding ith element $\beta_{n,i}(\hat{t}_T)$ in amplitude vector $\boldsymbol{\beta}_n(\hat{t}_T)$ denotes the actual target amplitude which is nonzero; however, other elements in $\boldsymbol{\beta}_n(\hat{t}_T)$, that is, $\beta_{n,s}(\hat{t}_T),\ s \ne i,\ 0 \le s \le S-1$, are zeros because the target Doppler is not equal to the frequencies $f_s^d \in F,\ s \ne i,\ 0 \le s \le S-1$. In a word, if the actual Doppler of target on $(\hat{t}_T, \mathbf{u}_n)$ corresponds to the ith frequency in F, that is, $f_n = f_i^d$, the amplitude vector $\boldsymbol{\beta}_n(\hat{t}_T)$ satisfies

$$\begin{aligned} \left|\beta_{n,s}(\hat{t}_T)\right| &\ne 0, \quad s = i, \\ \left|\beta_{n,s}(\hat{t}_T)\right| &= 0, \quad s \ne i,\ 0 \le s \le S-1. \end{aligned} \tag{3}$$

According to (3), there is not more than one nonzero element in $\boldsymbol{\beta}_n(\hat{t}_T)$; that is, $\|\boldsymbol{\beta}_n(\hat{t}_T)\|_0 \le 1$.

Regarding sparse scenario E, a sparse amplitude vector $\boldsymbol{\beta}(\hat{t}_T) = [\boldsymbol{\beta}_0^T(\hat{t}_T), \ldots, \boldsymbol{\beta}_{N-1}^T(\hat{t}_T)]^T \in \mathbb{C}^{SN}$ defined on joint angle-Doppler domain $\Omega \otimes F$ is employed, where "$\otimes$" denotes direct product. The nth element, $\boldsymbol{\beta}_n^T(\hat{t}_T),\ n = 0, \ldots, N-1$, in $\boldsymbol{\beta}(\hat{t}_T)$ denotes the amplitude for the target in the nth grid $\mathbf{u}_n \in \Omega$. If there actually exists a target in the nth grid, that is, $\mathbf{u}_n \in T$, its amplitude being $\boldsymbol{\beta}_n^T(\hat{t}_T),\ n \in I_T$ satisfies (3); otherwise, the amplitude vectors $\boldsymbol{\beta}_n^T(\hat{t}_T),\ n \notin I_T,\ n = 0, \ldots, N-1$, are equal to zero because there is not an actual target. Omitting the fast time term $\hat{t}_T$, we abbreviate $\boldsymbol{\beta}_n(\hat{t}_T)$ to $\boldsymbol{\beta}_n = [\beta_{n,0}, \ldots, \beta_{n,S-1}]^T,\ n = 0, \ldots, N-1$ and $\boldsymbol{\beta}(\hat{t}_T)$ to $\boldsymbol{\beta} = [\boldsymbol{\beta}_0^T, \ldots, \boldsymbol{\beta}_{N-1}^T]^T$. The amplitude vector $\boldsymbol{\beta}$ is also named as sparse target scenario in the paper which is a "block-structure" vector as follows:

$$\boldsymbol{\beta} = \left[\underbrace{\beta_{0,0}, \ldots, \beta_{0,S-1}}_{\text{block } 0\#}, \ldots, \underbrace{\beta_{N-1,0}, \ldots, \beta_{N-1,S-1}}_{\text{block } (N-1)\#}\right]^T. \tag{4}$$

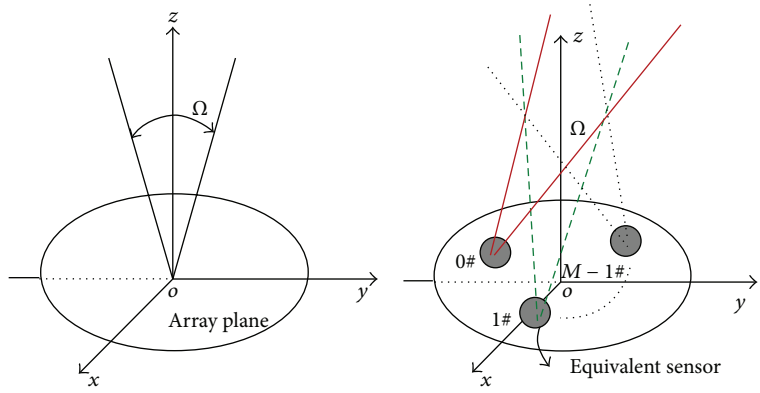


Figure 1: Illumination and receiving beam. (a) Illumination mode. (b) Receiving mode.

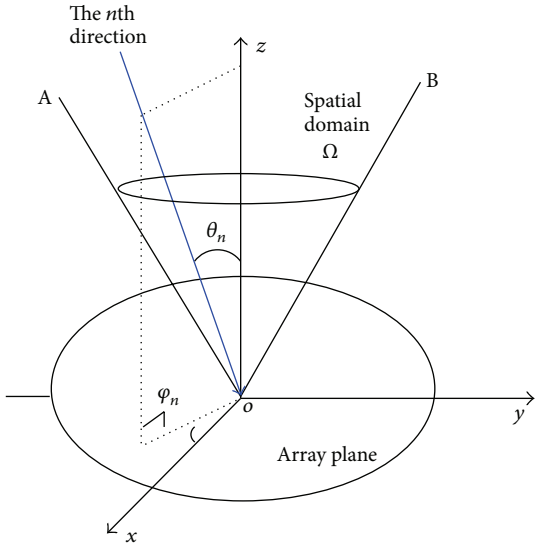


Figure 2: The geometry of array and target scenario.

The scenario $\boldsymbol{\beta}$ consists of N blocks in which each one has S elements. The nonzero element $\beta_{n,s} \in \boldsymbol{\beta}$, $n = 0, \ldots, N-1$; $s = 0, \ldots, S-1$ means that there is a target located in nth grid $\mathbf{u}_n$ and its Doppler is equal to sth frequency $f_s^d \in F$; otherwise, the zero element $\beta_{n,s} \in \boldsymbol{\beta}$ means that there is not a target corresponding to nth grid $\mathbf{u}_n$ and sth frequency $f_s^d \in F$. Therefore, there are not more than $\|\boldsymbol{\beta}\|_0 = K$ nonzero elements in $\boldsymbol{\beta}$ with respect to sparse scenario E.

With respect to an arbitrary target on $(\hat{t}_T, \mathbf{u}_n)$ with Doppler $f_s^d \in F$, the radar return of reference sensor $(0, 0, 0)$ during the reception of the rth PRI is

$$s_{n,s}^r(\hat{t}) = \beta_{n,s} p(\hat{t} - \hat{t}_T) \exp\left(j2\pi f_s^d \hat{t}\right) \exp\left(j2\pi f_s^d t_r\right), \quad (5)$$

where the superscript "r" corresponds to the rth PRT and the subscript "n, s" corresponds to the nth spatial grid $\mathbf{u}_n \in \Omega$ and the sth element $f_s^d \in F$. The receiving signal of the mth equivalent sensor about $\boldsymbol{\beta}$ which is assumed to be stationary during an observation period can be expressed as

$$y_m^r(\hat{t}) = \sum_{n=0}^{N-1} \sum_{s=0}^{S-1} s_{n,s}^r(\hat{t}) \cdot \exp\left[jk\left(u_n x_m + v_n y_m\right)\right], \quad (6)$$

where the subscript "m" represents the mth equivalent sensor and $k = 2\pi/\lambda$. The outputs of M equivalent sensors in the rth PRI can be shown as

$$\mathbf{y}^r(\hat{t}) = \mathbf{A} \times \mathbf{s}^r(\hat{t}), \quad (7)$$

where $\mathbf{y}^r(\hat{t}) = [y_0^r(\hat{t}), \ldots, y_{M-1}^r(\hat{t})]^T \in \mathbb{C}^M$ and $\mathbf{A} \in \mathbb{C}^{M \times SN}$ is the array response matrix about scenario $\boldsymbol{\beta}$ as follows:

$$\mathbf{A} = \left[\underbrace{\mathbf{a}_0, \ldots, \mathbf{a}_0}_{S} \cdots \underbrace{\mathbf{a}_{N-1}, \ldots, \mathbf{a}_{N-1}}_{S}\right]. \quad (8)$$

There are N blocks in $\mathbf{A}$ and each block contains S identical steering vectors. The nth steering vector $\mathbf{a}_n$ in $\mathbf{A}$ is

$$\mathbf{a}_n = [a_{0,n}, \ldots, a_{M-1,n}]^T, \quad (9)$$

where $a_{m,n} = \exp[jk(u_n x_m + v_n y_m)]$, $m = 0, \ldots, M-1$. In (7), $\mathbf{s}^r(\hat{t}) \in \mathbb{C}^{SN}$ is the return of $\boldsymbol{\beta}$ at the full time $(\hat{t}, t_r)$:

$$\mathbf{s}^r(\hat{t}) = \boldsymbol{\beta} \odot \mathbf{p}(\hat{t} - \hat{t}_T) \odot \mathbf{e}^r, \quad (10)$$

where "$\odot$" denotes Hadamard product. In (10), $\mathbf{p}(\hat{t} - \hat{t}_T) \in \mathbb{C}^{SN}$ represents the part corresponding to fast time and $\mathbf{e}^r \in \mathbb{C}^{SN}$ represents the part corresponding to slow time of returns. If $\tilde{\mathbf{p}}(\hat{t} - \hat{t}_T) \in \mathbb{C}^S$ denotes the fast time term as shown in (5) which can be expressed as

$$\begin{aligned}\tilde{\mathbf{p}}(\hat{t} - \hat{t}_T) &= p(\hat{t} - \hat{t}_T)\left[\exp\left(j2\pi f_0^d \hat{t}\right), \ldots, \exp\left(j2\pi f_{S-1}^d \hat{t}\right)\right]^T \\ &= \left[\tilde{p}_0(\hat{t} - \hat{t}_T), \ldots, \tilde{p}_{S-1}(\hat{t} - \hat{t}_T)\right]^T,\end{aligned} \quad (11)$$

then $\mathbf{p}(\hat{t}-\hat{t}_T)$ is N repetition of $\tilde{\mathbf{p}}(\hat{t}-\hat{t}_T)$ as follows:

$$\mathbf{p}\left(\hat{t}-\hat{t}_T\right)=\left[\underbrace{\tilde{\mathbf{p}}^T\left(\hat{t}-\hat{t}_T\right),\ldots,\tilde{\mathbf{p}}^T\left(\hat{t}-\hat{t}_T\right)}_{N}\right]^T. \tag{12}$$

Similarly, the item $\mathbf{e}^r$ in (10) is N repetitions of $\tilde{\mathbf{e}}^r$ as follows:

$$\mathbf{e}^r=\left[\underbrace{\left(\tilde{\mathbf{e}}^r\right)^T,\ldots,\left(\tilde{\mathbf{e}}^r\right)^T}_{N}\right]^T, \tag{13}$$

where $\tilde{\mathbf{e}}^r=[\exp(j2\pi f_0^d t_r),\ldots,\exp(j2\pi f_{S-1}^d t_r)]^T$ is the item corresponding to the slow time in (5).

Substituting (12) and (13) into (7), the item in $\mathbf{s}^r(\hat{t})$ corresponding to slow time can be transferred to measurement matrix $\mathbf{A}$ and a new measurement model can be formulated as

$$\mathbf{y}^r\left(\hat{t}\right)=\mathbf{A}^r\times\mathbf{s}\left(\hat{t}\right), \tag{14}$$

where $\mathbf{s}(\hat{t})\in\mathbb{C}^{SN}$ stems from $\mathbf{s}^r(\hat{t})$ after eliminating the Doppler item $\mathbf{e}^r$, which is the return of $\boldsymbol{\beta}$ in the 0th PRI. The item $\mathbf{s}(\hat{t})$ is named as the return of $\boldsymbol{\beta}$ and can be expressed as

$$\begin{aligned}&\mathbf{s}\left(\hat{t}\right)\\&=\mathbf{s}^r(\hat{t})\big|_{r=0}=\boldsymbol{\beta}\odot\mathbf{p}\left(\hat{t}-\hat{t}_T\right)\\&=\left[\underbrace{s_{0,0}\left(\hat{t}\right),\ldots,s_{0,S-1}\left(\hat{t}\right)}_{\text{block }0\#},\ldots,\underbrace{s_{N-1,0}\left(\hat{t}\right),\ldots,s_{N-1,S-1}\left(\hat{t}\right)}_{\text{block }N-1\#}\right]^T.\end{aligned} \tag{15}$$

It is seen that $\mathbf{s}(\hat{t})$ is also a sparse signal which shares the same support as target scenario $\boldsymbol{\beta}$; therefore, $\mathbf{s}(\hat{t})$ is also regarded as sparse target scenario. The sth element in the nth block $s_{n,s}(\hat{t})$ of $\mathbf{s}(\hat{t})$, $n=0,\ldots,N-1$, $s=0\ldots,S-1$ denotes the return corresponding to nth target in Ω on sth Doppler of dictionary F at fast time $\hat{t}$. The matrix $\mathbf{A}^r\in\mathbb{C}^{M\times SN}$ in (14) stems from immigration of $\mathbf{e}^r$ to $\mathbf{A}$ based on (7), which is a joint angle-Doppler representation basis as follows:

$$\begin{aligned}&\mathbf{A}^r\\&=\left[\underbrace{e^{j2\pi f_0^d t_r}\mathbf{a}_0,\ldots,e^{j2\pi f_{S-1}^d t_r}\mathbf{a}_0}_{\text{block }0\#},\ldots,\underbrace{e^{j2\pi f_0^d t_r}\mathbf{a}_{N-1},\ldots,e^{j2\pi f_{S-1}^d t_r}\mathbf{a}_{N-1}}_{\text{block }N-1\#}\right].\end{aligned} \tag{16}$$

The matrix $\mathbf{A}^r$ also has a block-structure, and there are N blocks in which each block has S columns. In the arbitrary nth block of $\mathbf{A}^r$, $n=0,\ldots,N-1$, the same one steering vector $\mathbf{a}_n$ provides a representation in spatial angle domain for the response of receiving array including M equivalent sensors, and the items $\exp(j2\pi f_s^d t_r)$, $s=0,\ldots,S-1$ provide the representation for the phases in Doppler domain with respect to S Doppler frequencies f_s^d, $s=0,\ldots,S-1$. Consequently, $\mathbf{A}^r$ is a representation for the return of target scenario $\boldsymbol{\beta}$ in joint angle-Doppler domain.

According to (7) and (14), the measurement $\mathbf{y}^r(\hat{t})$ about the return $\mathbf{s}^r(\hat{t})$ of $\boldsymbol{\beta}$ in arbitrary rth PRI can be reformulated as the measurement about the return $\mathbf{s}(\hat{t})=\mathbf{s}^r(\hat{t})|_{r=0}$ of $\boldsymbol{\beta}$ in the 0th PRI. The sparse scenario $\mathbf{s}(\hat{t})$ can be recovered using the measurement model as shown in (14) if sensing matrix $\mathbf{A}^r$ satisfies RIP.

A hardware structure having M receiver channels will be considered in the following. Owing to low SNR characteristic of returns in active radar, we resort to matching filter on the output of array $\mathbf{y}^r(\hat{t})$. The output after matching filter can be shown as

$$\mathbf{y}_{\mathrm{MF}}^r\left(\hat{t}\right)=\mathrm{MF}\left\{\mathbf{y}^r\left(\hat{t}\right)\right\}=\mathrm{MF}\left\{\mathbf{A}^r\times\mathbf{s}\left(\hat{t}\right)\right\}=\mathbf{A}^r\times\mathrm{MF}\left\{\mathbf{s}\left(\hat{t}\right)\right\}, \tag{17}$$

where $\mathrm{MF}\{\cdot\}$ denotes matching filter operation. Taking additive noise $\mathbf{n}_{\mathrm{MF}}^r(\hat{t})\in\mathbb{C}^M$ into account, the measurement model after matching filter can be expressed as

$$\mathbf{y}_{\mathrm{MF}}^r\left(\hat{t}\right)=\mathbf{A}^r\times\mathbf{s}_{\mathrm{MF}}\left(\hat{t}\right)+\mathbf{n}_{\mathrm{MF}}^r\left(\hat{t}\right), \tag{18}$$

where the return $\mathbf{s}_{\mathrm{MF}}(\hat{t})=\mathrm{MF}\{\mathbf{s}(\hat{t})\}=\mathrm{MF}\{\boldsymbol{\beta}\odot\mathbf{p}(\hat{t}-\hat{t}_T)\}$ of $\boldsymbol{\beta}$ after matching filter can be expressed as

$$\mathbf{s}_{\mathrm{MF}}\left(\hat{t}\right)=\boldsymbol{\beta}\odot\mathrm{MF}\left\{\mathbf{p}\left(\hat{t}-\hat{t}_T\right)\right\}, \tag{19}$$

where $\mathrm{MF}\{\mathbf{p}(\hat{t}-\hat{t}_T)\}$ is the output of matching filter about $\mathbf{p}(\hat{t}-\hat{t}_T)$, which is abbreviated to $\mathbf{p}^{mf}(\hat{t}-\hat{t}_T)\in\mathbb{C}^{SN}$ here. According to the expression of $\mathbf{p}(\hat{t}-\hat{t}_T)$ in (12), $\mathbf{p}^{mf}(\hat{t}-\hat{t}_T)$ can be shown as

$$\mathbf{p}^{mf}\left(\hat{t}-\hat{t}_T\right)=\left[\underbrace{\left(\tilde{\mathbf{p}}^{mf}(\hat{t}-\hat{t}_T)\right)^T,\ldots,\left(\tilde{\mathbf{p}}^{mf}(\hat{t}-\hat{t}_T)\right)^T}_{N}\right]^T, \tag{20}$$

where $\tilde{\mathbf{p}}^{mf}(\hat{t}-\hat{t}_T)\in\mathbb{C}^S$ is $\mathrm{MF}\{\tilde{\mathbf{p}}(\hat{t}-\hat{t}_T)\}$ as follows:

$$\begin{aligned}\tilde{\mathbf{p}}^{mf}\left(\hat{t}-\hat{t}_T\right)&=\left[\mathrm{MF}\left\{\tilde{p}_0\left(\hat{t}-\hat{t}_T\right)\right\},\ldots,\mathrm{MF}\left\{\tilde{p}_{S-1}\left(\hat{t}-\hat{t}_T\right)\right\}\right]^T\\&=\left[p_0^{mf}\left(\hat{t}-\hat{t}_T\right),\ldots,p_{S-1}^{mf}\left(\hat{t}-\hat{t}_T\right)\right]^T.\end{aligned} \tag{21}$$

Due to the sparse characteristic of $\boldsymbol{\beta}$, the return $\mathbf{s}_{\mathrm{MF}}(\hat{t})$ is also a sparse vector with the same support as $\boldsymbol{\beta}$.

Considering the return with maximum SNR after matching filter appears at the fast time $\hat{t}_{T-\mathrm{MF}}$ with respect to target at range $\hat{t}_T$, the return $\mathbf{s}_{\mathrm{MF}}(\hat{t})$ at $\hat{t}_{T-\mathrm{MF}}$ can be used as the measurement to recover the unknown sparse scenario. The version of model (18) at $\hat{t}_{T-\mathrm{MF}}$ can be shown as

$$\mathbf{y}_{\mathrm{MF}}^r=\mathbf{A}^r\times\mathbf{s}+\mathbf{n}_{\mathrm{MF}}^r, \tag{22}$$

where $\mathbf{y}_{\mathrm{MF}}^r=\mathbf{y}_{\mathrm{MF}}^r(\hat{t}_{T-\mathrm{MF}})$ and $\mathbf{n}_{\mathrm{MF}}^r=\mathbf{n}_{\mathrm{MF}}^r(\hat{t}_{T-\mathrm{MF}})$. The signal $\mathbf{s}\in\mathbb{C}^{SN}$ is the abbreviation of $\mathbf{s}_{\mathrm{MF}}(\hat{t}_{T-\mathrm{MF}})$ as follows:

$$\mathbf{s}=\mathbf{s}_{\mathrm{MF}}\left(\hat{t}_{T-\mathrm{MF}}\right)=\left[\underbrace{s_{0,0},\ldots,s_{0,S-1}}_{\text{block }0\#},\ldots,\underbrace{s_{N-1,0},\ldots,s_{N-1,S-1}}_{\text{block }N-1\#}\right]^T. \tag{23}$$

The element $s_{n,s}$, $n=0,\ldots,N-1$, $s=0,\ldots,S-1$, is the sth element in the nth block of signal $\mathbf{s}$, which can be written as $s_{n,s}=\beta_{n,s}p_s^{mf}(\hat{t}-\hat{t}_T)|_{\hat{t}=\hat{t}_{T-\mathrm{MF}}}$. Due to the same sparse characteristic between $\mathbf{s}$ and $\boldsymbol{\beta}$, the signal $\mathbf{s}$ can also be seen as target sparse scenario in the paper.

The measurement model in (22) is a SMV model in CS framework; sparse scenario $\mathbf{s}$ could be reconstructed via the solution as shown in (1) when the number of measurements M is sufficient and SNR is sufficiently high. During an observation period, the R measurements $\{\mathbf{y}_{\mathrm{MF}}^r\}_{r=0}^{R-1}$ at the same fast time $\widehat{t}_{T-\mathrm{MF}}$ in R pulse repetition intervals satisfy MMV measurement model and then can be used to recover sparse target scenario $\mathbf{s}$ through MUSIC, SOMP [7], and CS-MUSIC [14]. What should be highlighted is that the model in (22) is developed based on multireceiver structure which needs the same number of receivers as receiving array sensors. This multireceiver structure is more expensive in price and larger in volume than traditional receiver structure in active radar. In the next section, the proposed strategy which employs multisensor single-receiver structure is presented to recover the sparse target scenario.

2.2. Measurement Model Using Single-Receiver. According to (14), there are R measurements about a fixed return $\mathbf{s}(\widehat{t})$ with respect to target scenario $\boldsymbol{\beta}$ at the same fast time $\widehat{t}$ during an observation period. In CS framework, few measurements about a sparse signal using random waveforms can be used to recover the original sparse signal with overwhelming probability. Inspired by this mechanism, the outputs of M subarray $\mathbf{y}^r(\widehat{t})$ can be randomly weighted through M phase shifters and then be summarized using single-receiver channel. The random weighted and summarized version of $\mathbf{y}^r(\widehat{t})$ is shown as follows:

$$z^r\left(\widehat{t}\right) = \left(\mathbf{f}^r\right)^T \mathbf{A}^r \mathbf{s}\left(\widehat{t}\right), \quad r = 0, \ldots, R-1, \tag{24}$$

where $\mathbf{f}^r = [\phi_0^r, \ldots, \phi_{M-1}^r]^T \in \mathbb{C}^M$, $r = 0, \ldots, R-1$, is the random weight in rth PRI. The weights remain unchanged in a pulse repetition period but take different values in different period. The element ϕ_m^r in $\mathbf{f}^r$ may be a random Bernoulli variable; for example, $\phi_m^r = \{\pm 1, \text{w.p. } 1/2\}$, $m = 0, \ldots, M-1$.

In principle, the random weighing onto $\mathbf{y}^r(\widehat{t})$ should be put into practice using M additional phase shifters connecting to the outputs of M subarrays. Assuming that B array elements belonging to mth subarray, $m = 0, \ldots, M-1$, can form the beam pointing to Ω when phase shifters of B corresponding transmitter and receiver (T/R) modules are set up as $C'_m = \{C'_{m,b}\}_{b=0}^{B-1}$ during the reception, the random weighting procedure as shown in (24) can be accomplished if phased shifters of T/R modules change their setup as $C''_m = \phi_m^r C'_m = \{\phi_m^r C'_{m,b}\}_{b=0}^{B-1}$. Accordingly, there is no requirement of additional phase shifters to achieve the random weighting onto the outputs of equivalent sensors.

According to (24), the measuring model with respect to the fixed return $\mathbf{s}(\widehat{t})$ of target scenario $\boldsymbol{\beta}$ can be formulated as

$$\begin{bmatrix} z^0\left(\widehat{t}\right) \\ \vdots \\ z^{R-1}\left(\widehat{t}\right) \end{bmatrix} = \begin{bmatrix} \left(\mathbf{f}^0\right)^T \times \mathbf{A}^0 \\ \vdots \\ \left(\mathbf{f}^{R-1}\right)^T \times \mathbf{A}^{R-1} \end{bmatrix} \times \mathbf{s}\left(\widehat{t}\right). \tag{25}$$

The above model can be rewritten as

$$\mathbf{z}\left(\widehat{t}\right) = \boldsymbol{\Phi} \times \mathbf{A}^s \times \mathbf{s}\left(\widehat{t}\right), \tag{26}$$

where $\mathbf{z}(\widehat{t}) = [z^0(\widehat{t}), \ldots, z^{R-1}(\widehat{t})]^T$ is the measurements about $\mathbf{s}(\widehat{t})$ through random weighting, and $\boldsymbol{\Phi} \in \mathbb{C}^{R \times RM}$ is the measurement matrix with block-diagonal structure as follows:

$$\begin{aligned} \boldsymbol{\Phi} &= \mathrm{diag}\left(\left(\mathbf{f}^0\right)^T, \ldots, \left(\mathbf{f}^{R-1}\right)^T\right) \\ &= \begin{bmatrix} \left(\mathbf{f}^0\right)^T & & \mathbf{0} \\ & \ddots & \\ \mathbf{0} & & \left(\mathbf{f}^{R-1}\right)^T \end{bmatrix}. \end{aligned} \tag{27}$$

The matrix $\mathbf{A}^s \in \mathbb{C}^{RM \times SN}$ is a joint angle-Doppler representation basis, which is a stack of R matrices as follows:

$$\mathbf{A}^s = \left[\left(\mathbf{A}^0\right)^T, \ldots, \left(\mathbf{A}^{R-1}\right)^T\right]^T = \begin{bmatrix} \mathbf{A}^0 \\ \vdots \\ \mathbf{A}^{R-1} \end{bmatrix}, \tag{28}$$

where the element $\mathbf{A}^r$, $r = 0, \ldots, R-1$, is shown as (16).

The summarization after random weighing as shown in (24) should also be processed via matching filter to improve SNR. Substituting (15), (16), and (19) into (24), a version of $z^r(\widehat{t})$ after matching filter can be expressed as follows:

$$\begin{aligned} & z_{\mathrm{MF}}^r\left(\widehat{t}\right) \\ &= \mathrm{MF}\left\{z^r\left(\widehat{t}\right)\right\} \\ &= \mathrm{MF}\left\{\left[\underbrace{e^{j2\pi f_0^d t_r}\left(\mathbf{f}^r\right)^T \times \mathbf{a}_0, \ldots, e^{j2\pi f_{S-1}^d t_r}\left(\mathbf{f}^r\right)^T \times \mathbf{a}_0}_{\text{block } 0\#}, \ldots, \right.\right. \\ &\qquad \left.\left.\underbrace{e^{j2\pi f_0^d t_r}\left(\mathbf{f}^r\right)^T \times \mathbf{a}_{N-1}, \ldots, e^{j2\pi f_{S-1}^d t_r}\left(\mathbf{f}^r\right)^T \times \mathbf{a}_{N-1}}_{\text{block } N-1\#}\right] \right. \\ &\qquad \left. \times\, \mathbf{s}(t)\right\} \\ &= \sum_{n=0}^{N-1} \sum_{s=0}^{S-1} \left\{e^{j2\pi f_s^d t_r}\left(\mathbf{f}^r\right)^T \times \mathbf{a}_n\right\} \cdot \mathrm{MF}\left\{s_{n,s}\left(\widehat{t}\right)\right\} \\ &= \left(\left(\mathbf{f}^r\right)^T \times \mathbf{A}^r\right) \times \mathbf{s}_{\mathrm{MF}}\left(\widehat{t}\right), \quad r = 0, \ldots, R-1. \end{aligned} \tag{29}$$

The measuring model corresponding to (26) after matching filter can be obtained based on (29) as follows:

$$\mathbf{z}_{\mathrm{MF}}\left(\widehat{t}\right) = \boldsymbol{\Phi} \times \mathbf{A}^s \times s_{\mathrm{MF}}\left(\widehat{t}\right) + \mathbf{n}_{\mathrm{MF}}\left(\widehat{t}\right), \tag{30}$$

where $\mathbf{n}_{\mathrm{MF}}(\widehat{t}) \in \mathbb{C}^R$ is the additive noise of measurement and $\mathbf{z}_{\mathrm{MF}}(\widehat{t}) = \mathrm{MF}\{\mathbf{z}(\widehat{t})\}$ can be expressed as $[z_{\mathrm{MF}}^0(\widehat{t}), \ldots, z_{\mathrm{MF}}^{R-1}(\widehat{t})]^T$ which is a R-dimension measurement.

As described above, the measurement at $\widehat{t}_{T-\mathrm{MF}}$ of $\mathbf{z}_{\mathrm{MF}}(\widehat{t})$ where the return after matching filter has maximum SNR can be shown as

$$\mathbf{z} = \boldsymbol{\Phi} \times \mathbf{A}^s \times \mathbf{s} + \mathbf{n}, \tag{31}$$

where $\mathbf{z} = \mathbf{z}_{\mathrm{MF}}(\widehat{t}_{T-\mathrm{MF}})$; $\mathbf{s} = \mathbf{s}_{\mathrm{MF}}(\widehat{t}_{T-\mathrm{MF}})$ is expressed in (23) which can be seen as sparse target scenario; $\mathbf{n} = \mathbf{n}_{MF}(\widehat{t}_{T-\mathrm{MF}})$

is additive Gaussian noise and $\mathrm{Re}(\mathbf{n})$, $\mathrm{Im}(\mathbf{n}) \sim N(0, \sigma^2\mathbf{I}_R)$, where $\mathrm{Re}(\mathbf{n})$ and $\mathrm{Im}(\mathbf{n})$ denote real and imaginary parts, respectively; $\mathbf{I}_R$ denotes R-dimension identity matrix. For the convenience of analysis, $\mathbf{D} = \mathbf{\Phi}\mathbf{A}^s$ is denoted as equivalent dictionary, and then (31) can be rewritten as

$$\mathbf{z} = \mathbf{D} \times \mathbf{s} + \mathbf{n}. \tag{32}$$

Dictionary $\mathbf{D}$ can be expressed as

$$\mathbf{D} = \left[\underbrace{\mathbf{d}_{0,0}, \ldots, \mathbf{d}_{0,S-1}}_{\text{block } 0\#}, \ldots, \underbrace{\mathbf{d}_{N-1,0}, \ldots, \mathbf{d}_{N-1,S-1}}_{\text{block } N-1\#} \right], \tag{33}$$

where $\mathbf{d}_{n,s}, n = 0, \ldots, N-1;\ s = 0, \ldots, S-1$, denotes the s-column in nth block of $\mathbf{D}$. The SNR of the return $s_{n,s} \in \mathbf{s}$ after matching filter about target $\beta_{n,s} \in \boldsymbol{\beta}$ is defined as

$$\mathrm{SNR}_{n,s} = \frac{\left|s_{n,s}\right|^2 \cdot \left\|\mathbf{d}_{n,s}\right\|_2^2}{R \cdot \left(2\sigma^2\right)}, \tag{34}$$

where $n = 0, \ldots, N-1,\ s = 0, \ldots, S-1$.

As shown in (31), we develop a measurement model with respect to sparse target scenario $\mathbf{s}$ (equivalent to $\boldsymbol{\beta}$) using a joint angle-Doppler representation, which needs only one receiver channel. The measurement is an SMV model in CS framework in which the equivalent dictionary $\mathbf{D}$ satisfies RIP when the number of measurement R is sufficient; therefore, the estimated target scenario $\widehat{\mathbf{s}} \in \mathbb{C}^{SN}$ can be recovered via the solution as shown in (1). The reconstructed $\widehat{\mathbf{s}}$ can be rewritten as $\widehat{\mathbf{s}} = [\widehat{\mathbf{s}}_0^T, \ldots, \widehat{\mathbf{s}}_{N-1}^T]^T$ in which element $\widehat{\mathbf{s}}_n = [\widehat{s}_{n,0}, \ldots, \widehat{s}_{n,S-1}]^T,\ n = 1, \ldots, N-1$ denotes the amplitude vector on the Doppler dictionary F of target located in the nth spatial grid of Ω. Integrating energies distributed in all Doppler elements on F for every nth target in Ω, a recovered target scenario defined on Ω can be shown as follows:

$$\widehat{\mathbf{s}}_\Omega = \left[\left\|\widehat{\mathbf{s}}_0\right\|_2, \ldots, \left\|\widehat{\mathbf{s}}_{N-1}\right\|_2\right]^T. \tag{35}$$

3. Sparse Target Scenario Recovery

In Section 2, the Doppler dictionary F is assumed as a known set during the generation of $\mathbf{A}^r$; in the practice, F is unknown because relative velocities between targets and radar are not priori. In this section, F is built using radar returns and then matrices $\mathbf{A}^r$ and $\mathbf{A}^s$ are presented; in the end, target scenario can be recovered based on measurement model (31).

3.1. Estimation of Targets' Doppler. A PAR should work on a conventional mode with one illumination and one receiving beam via appropriate setup on phase shifters. The radar returns including R' pulse in one observation period with respect to K targets distributed in Ω; at the same range, $\widehat{t}_T$ can be expressed as follows:

$$q^r\left(\widehat{t}\right) = \sum_{k=0}^{K-1} \beta_k p\left(\widehat{t} - \widehat{t}_T\right) \exp\left[j2\pi x_k\left(\widehat{t} + t_r\right)\right], \quad r = 0, \ldots, R'-1, \tag{36}$$

where $X = \{x_k, k = 0, \ldots, K-1\}$ denotes the actual targets' Doppler set and $\beta_k,\ k = 0, \ldots, K-1$ denotes the kth target amplitude. The outputs after matching filter about the return in (36) at the same fast time $\widehat{t} = \widehat{t}_{T-\mathrm{MF}}$ where maximum SNR level exists are

$$q^r = \sum_{k=0}^{K-1} \beta_k^{mf} \exp\left(j2\pi x_k t_r\right), \quad r = 0, \ldots, R'-1, \tag{37}$$

where $\beta_k^{mf},\ k = 0, \ldots, K-1$, is the corresponding target amplitude after matching filter. According to (37), the returns at the same fast time $\widehat{t}_{T-\mathrm{MF}}$ during R' pulse repetition periods are linear combination of K complex sinusoids on the slow time dimension.

Due to the sparse consideration about target scenario, the number of complex sinusoids in (37) is far less than the number of the frequencies in set F_B which contains all possible Doppler frequencies in radar returns when the pulse repetition frequency is $F_r = 1/T_r$. The set F_B stems from discretizing $[0, F_r]$ with an interval Δf as follows:

$$F_B = \left\{f_b = b \cdot \Delta f, b = 0, \ldots, B-1\right\}, \tag{38}$$

where $B = F_r/\Delta f, B \gg K$. In consequence, X could be reconstructed using CS framework. A sparse target scenario $\boldsymbol{\alpha} \in \mathbb{C}^B$ defined on F_B can be expressed as

$$\boldsymbol{\alpha} = \left[\alpha_0, \ldots, \alpha_{B-1}\right]^T, \tag{39}$$

where $\alpha_b = \beta_k^{mf}$ when $f_b = x_k$; otherwise $\alpha_b = 0$ for $b = 0, \ldots, B-1$ and $k = 0, \ldots, K-1$. Consequently, there are K nonzero elements in $\boldsymbol{\alpha}$. The returns as shown in (37) can be reformulated as

$$\mathbf{q} = \mathbf{E}\boldsymbol{\alpha} + \mathbf{n}, \tag{40}$$

where $\mathbf{q} = [q^0, \ldots, q^{R'-1}]^T \in \mathbb{C}^{R'}$ is the measurement; $\mathbf{n} \in \mathbb{C}^{R'}$ is additive noise; and $\mathbf{E} \in \mathbb{C}^{R' \times B}$ is the representation basis in frequency domain as follows:

$$\mathbf{E} = \left[\mathbf{e}(0), \ldots, \mathbf{e}(B-1)\right]. \tag{41}$$

The column $\mathbf{e}(b) \in \mathbb{C}^{R'},\ b = 0, \ldots, B-1$, of $\mathbf{E}$ can be shown as

$$\mathbf{e}(b) = \left[\exp\left(j2\pi f_b t_0\right), \ldots, \exp\left(j2\pi f_b t_{R'-1}\right)\right]^T, \tag{42}$$

where $f_b = b \cdot \Delta f$ and $t_r = r \cdot T_r,\ r = 0, \ldots, R'-1$.

Considering measurement model (40), measuring matrix $\mathbf{E}$ is a redundant dictionary in frequency domain. The sparse signal $\boldsymbol{\alpha}$ can be recovered using $R' \ll B$ measurements in CS framework [17]. Denoting $\widehat{\boldsymbol{\alpha}} = [\widehat{\alpha}_0, \ldots, \widehat{\alpha}_{B-1}]$ as the reconstructed signal, the estimated targets' Doppler set can be obtained when we detect signals in $\widehat{\boldsymbol{\alpha}}$ through an appropriate threshold T_h. Assuming that the signal indices in $\widehat{\boldsymbol{\alpha}}$ where the magnitude is larger than T_h is $\Omega_b = \{b \mid |\widehat{\alpha}_b| \geq T_h, b = 0, \ldots, B-1\}$, then the estimated target Doppler $\widehat{X}$ can be shown as follows if we rewrite Ω_b as $\Omega_b = \{b(k), k = 0, \ldots, \widehat{K}-1\}$:

$$\widehat{\mathrm{X}} = \left\{ f_{b(k)} = b(k)\,\Delta f, b(k) \in \Omega_b \right\}, \tag{43}$$

where $\widehat{K}$ is the element number of Ω_b.

3.2. Generation of Doppler Dictionary. In Section 3.1, an estimated targets' Doppler set $\widehat{X}$ defined on discretized set F_B is presented. In general, the actual target Doppler $x_k \in X$ does not accurately lie on the grid of F_B; hence, x_k can be expressed as $x_k = b(k)\Delta f + \Delta x_k$ where $|\Delta x_k| \leq 0.5\Delta f$ is the error between x_k and its approximating frequency $b(k)\Delta f, b(k) \in \Omega_b$ in F_B. When the error term Δx_k is large enough, more than one frequency components will appear in $\widehat{X}$ to approximate actual Doppler x_k. For example, if the error $\Delta x_k = 0.5\Delta f$, that is, $x_k = b(k) \cdot \Delta f + 0.5\Delta f$, two frequencies $b(k)\Delta f$ and $(b(k)+1)\Delta f$ whose magnitude exceed detection threshold may arise in $\widehat{X}$. In conclusion, there are usually more frequency components in the estimated Doppler set $\widehat{X}$ than actual set X, that is, $\widehat{K} \geq K$.

With respect the measurement model (31), sparse scenario $\boldsymbol{\beta}$ could be recovered with overwhelming probability when the number of measurement $R \geq C\mu^2(\boldsymbol{\Phi}, \mathbf{A}^s)K\log(SN)$ [2], where $\mu(\boldsymbol{\Phi}, \mathbf{A}^s)$ is the mutual coherence between $\boldsymbol{\Phi}$ and $\mathbf{A}^s$. Therefore, while R and N are fixed; then, the number of frequencies in dictionary F must satisfy $S \leq S_{\max}$ where $S_{\max}$ denotes the maximum numerical value which assure that the measurement matrix in (31) satisfy RIP. In summary, the frequency number in dictionary F should satisfy $\widehat{K} \leq S \leq S_{\max}$. The Doppler dictionary F can be generated based on $\widehat{X}$ in order to include all possible components in actual Doppler set X. Generally, F may be formulated as

$$F = \widehat{X}. \tag{44}$$

In the case where the component number of $\widehat{X}$ is small, we can appropriately extend dictionary F based on $\widehat{X}$. For example, there is a neighbor frequency whose amplitude is close to but not larger than detection threshold T_h, so we can put this neighbor frequency into $\widehat{X}$ in order that dictionary F includes all actual Doppler in X as far as possible.

3.3. Resolution of Doppler Dictionary. The Doppler dictionary F which is employed to produce joint angle-Doppler representation $\mathbf{A}^s$ comes from the estimated targets' Doppler set $\widehat{X}$ in which the frequency resolution depends on the resolution of F_B; therefore the resolution of F is Δf. As expressed in (28), the representation basis $\mathbf{A}^s$ is the stack of multiple $\mathbf{A}^r$. The $S \cdot N$ columns in $\mathbf{A}^s$ can be divided into N blocks in which each one comprises S subcolumns. According to (16) and (28), the (n,s)th column $\mathbf{a}^s_{n,s} \in \mathbb{C}^{SN}$ in $\mathbf{A}^s\, \forall n \in [0, N-1],\ s \in [0, S-1]$, which denotes the sth subcolumn of nth block, can be expressed as follows:

$$\mathbf{a}^s_{n,s} = \left[\exp\left(j2\pi f_s t_0\right)\mathbf{a}_n^T, \ldots, \exp\left(j2\pi f_s t_{R-1}\right)\mathbf{a}_n^T\right]^T. \tag{45}$$

In $\mathbf{a}^s_{n,s}$, the identical items $\mathbf{a}_n$ models for the relationship of spatial phases embedded in the M subarrays output due to a target located in $\mathbf{u}_n \in \Omega$ in spatial domain, and the items $\exp(j2\pi f_s t_r),\ r = 0, \ldots, R-1$, models for phase relationship existing in returns of all subarrays due to a target having Doppler $f_s \in F$ on the slow time dimension.

As addressed above, there usually is an estimating error Δx_k between actual Doppler x_k and its estimation $f_s = b(k)\Delta f \in F$ as shown in $x_k = b(k)\Delta f + \Delta x_k$. During an observation period including R pulses, the actual phase history in the slow time domain is $\{\exp(j2\pi x_k t_r), r = 0, \ldots, R-1\}$, while the phase history in the column $\mathbf{a}^s_{n,s}$ corresponding to sth components $f_s \in F$ is $\{\exp(j2\pi f_s t_r), r = 0, \ldots, R-1\}$. The phase error ΔP between the above two phase histories is $\Delta P = 2\pi \cdot \Delta x_k \cdot (t_{R-1} - t_0)$. In order to reduce the loss due to the phase error ΔP, we impose a limit $\Delta P < \pi/4$ for all possible estimating errors Δx_k. Considering the maximum value of $\Delta x_k = 0.5\Delta f$, the resolution Δf in F should satisfy

$$\Delta f < \frac{1/4}{t_{R-1} - t_0}, \tag{46}$$

where $(t_{R-1} - t_0) = R \cdot T_r$ is the time of duration during in an observation period as shown in model (31).

3.4. The Choice on the Measurement Number. In CS framework, more measurements can bring better recovery performance; therefore, the measurement number R in model (31) should be as large as possible. As described above, R is the number of pulses during a radar observation, so there are 3 factors about R being taken into account in the practice. Firstly, because the maximum moving distance between the radar platform and targets should not be longer than a radar range cell during an observation, so the choice of R should satisfy inequality $v_{\max} \cdot (RT_r) \leq r_{\text{cell}}$ where $v_{\max}$ denotes the maximum relative velocity between radar and targets and r_{cell} denotes the range cell of radar. Secondly, the increment of measurement number R brings more computational complexity in the procedure of CS recovery; therefore, the value of R should not be too large in order that the digital signal processor (DSP) of the radar can meet the computational demand. In general, the choice of R should assure that CS recovery can be finished in one observation period. Lastly, the measurement number R must satisfy (46); that is, $R < (1/4)/(\Delta f \cdot T_r)$, which assures that the phase error ΔP between the actual and the represented phase history cannot be more than $\pi/4$. In summary, the value of R should be as large as possible after the consideration on the above 3 factors.

For example, if the range cell of radar is $r_{\text{cell}} = 5$ m, the PRI is $T_r = 100\,\mu$s, and the maximum relative velocity between

radar and targets is $v_{\max} = 100\,\mathrm{m/s}$; the measurement number needs to satisfy $R \le r_{\mathrm{cell}}/(v_{\max} \cdot T_r) = 500$ according to the first factor in the above. If the resolution of Doppler dictionary F is set as $\Delta f = 25\,\mathrm{Hz}$, the measurement number should satisfy $R < (1/4)/(\Delta f \cdot T_r) = 100$ based on the factor 3. Supposing that the computation power of the DSP of radar is sufficient to accomplish the recovery procedure, the choice on the measurement number should satisfy $R < 100$ after consideration on the factor 1 and the factor 3.

3.5. Sparse Target Scenario Reconstruction. Based on the above description, we could reconstruct the sparse target scenario through the procedure as follows.

(1) Select the appropriate resolution of F according to (46) and then build the discretized frequency set F_B in (38).

(2) Setup radar on the mode of one illumination and one receiving beam. Get the estimated Doppler set $\widehat{X}$ in (43) from the measurement model in (40).

(3) Produce the Doppler dictionary F according to (44).

(4) Setup radar on the mode of multisensor single-receiver. Get the recovered $\hat{\mathbf{s}}$ according to measurement model (31) and then the target scenario $\hat{\mathbf{s}}_\Omega$ in Ω from (35).

The toolbox CVX [18] is employed to reconstruct $\boldsymbol{\alpha}$ in measurement model (40) and $\mathbf{s}$ in (31) according to the solution as shown in (1).

4. Numerical Experiments

In the section, the performance of the proposed strategy is evaluated through numerical experiments and the algorithm MUSIC is employed as a benchmark for comparison. What should be noted is that the proposed strategy as shown in (31) needs only one receiver channel, while the MUSIC algorithm is based on the measuring model as shown in (22) which needs multiple receiver channels. That is, the proposed strategy takes a more economical way than the benchmark algorithm MUSIC on the measurement about target scenario.

A PAR with wavelength $\lambda = 0.02\,\mathrm{m}$ is considered in all experiments. The shape of array plane is a circle with diameter 12.5λ, and all array sensors are partitioned into $M = 8$ subarrays which are randomly distributed in the array plane. The interest target area $\Omega = \{|\theta| \le 3^\circ, \varphi \in [0, 2\pi]\}$ is uniformly discretized into $N = 37$ grids. The radar PRI is $100\,\mu\mathrm{s}$, the range cell of radar is set as $r_{\mathrm{cell}} = 5\,\mathrm{m}$, and the relative velocity between radar and targets is not more than $v_{\max} = 100\,\mathrm{m/s}$. An experiment point $\mathrm{St} = \{K, \mathrm{SNR}\}$ is defined when target numbers K and SNRare fixed. With respect to each point St, 1000 times of independent numerical experiments are executed to evaluate the recovery performance.

4.1. The Procedure of One Experiment. The procedure of one numerical experiment will be illustrated using an example. In the example, a sparse target scenario where two adjacent targets are randomly distributed in Ω is investigated. The SNRs of two targets are both equal to 3 dB, and their Doppler sets are 3765 Hz and 3870 Hz, respectively. The procedure of the proposed strategy is shown as follows.

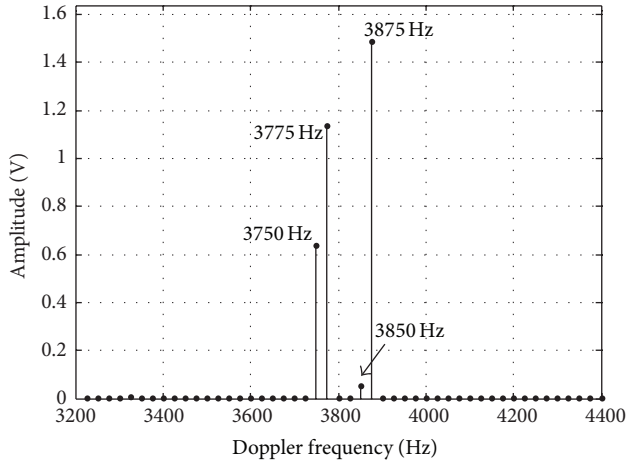


Figure 3: Recovered target scenario $\widehat{\boldsymbol{\alpha}}$ in Doppler domain.

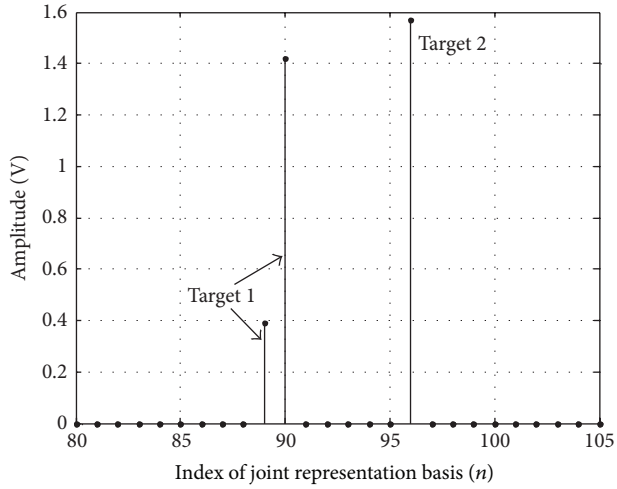


Figure 4: Recovered target scenario $\hat{\mathbf{s}}$ in joint angle-Doppler domain.

(1) Firstly, the number of measurements in model (31) is set up as $R = 80$ which will not bring the migration across a range cell r_{cell} during a radar observation period as illustrated in Section 3.4, so, according to (46), the resolution of dictionary F should satisfy $\Delta f < 0.25/(RT_r)$; that is, $\Delta f < 31$ Hz. In the experiment, the resolution Δf is set to 25 Hz. (2) According to model (37), sparse target scenario $\widehat{\boldsymbol{\alpha}}$ in Doppler domain is achieved using $R' = 100$ returns. The signal $\widehat{\boldsymbol{\alpha}}$ is a B-dimension vector where $B = F_r/\Delta f = 400$ and one part of it is depicted as Figure 3.

As shown in Figure 3, the estimated Doppler set $\widehat{X}$ is $\{3750, 3775, 3875\}$Hz after detection. There are 3 numbers of elements in $\widehat{X}$ which are larger than the real number of targets; that is, $(|\widehat{\mathrm{X}}| = 3) > (K = 2)$ because actual target Doppler $\{x_1, x_2\}$ does not accurately lie on the

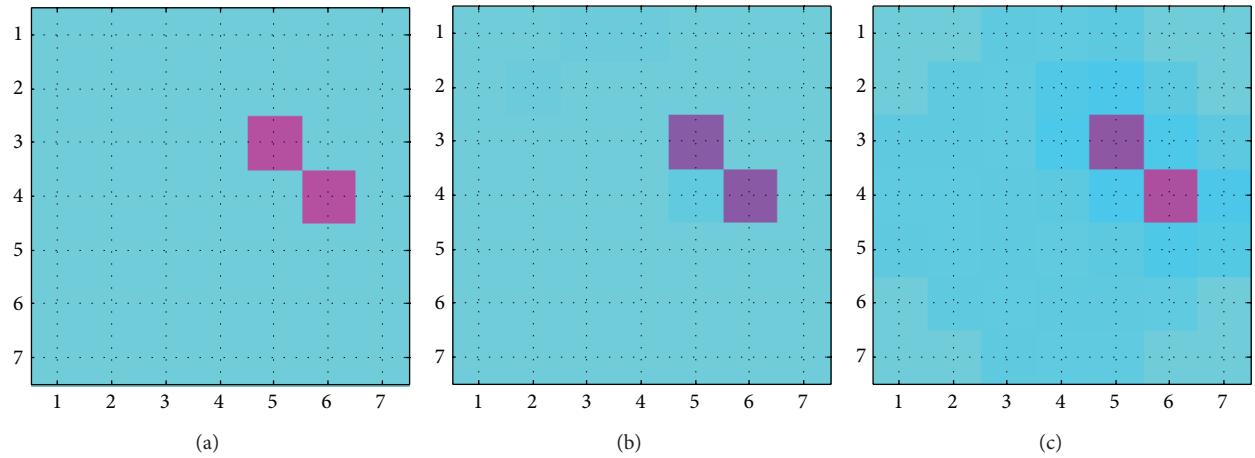


Figure 5: Recovered target scenario with two adjacent targets. (a) Original target scenario. (b) Recovery of the proposed strategy. (c) Recovery of MUSIC.

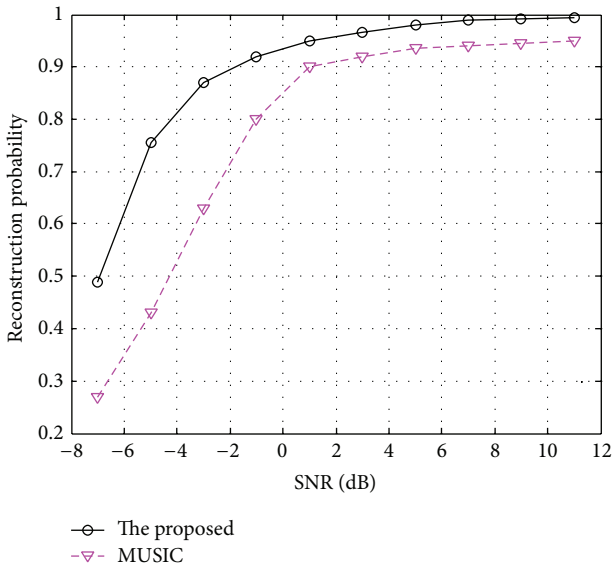


Figure 6: Reconstruction probability with two adjacent targets.

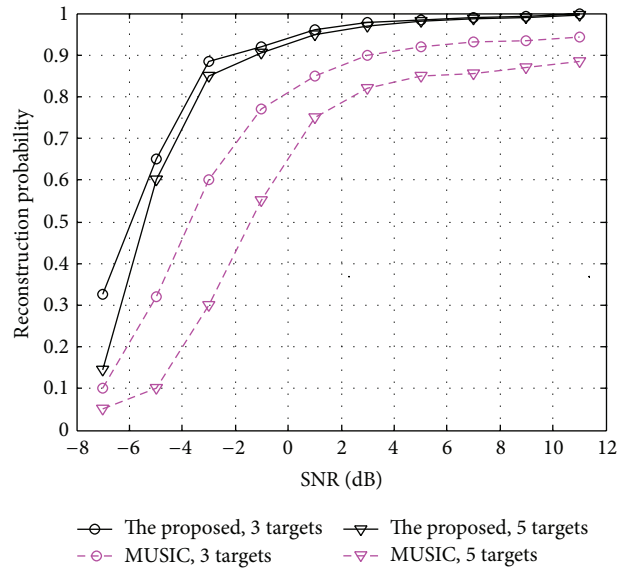


Figure 7: Reconstruction probability with multiple randomly distributed targets.

grid of discretized Doppler set F_B, where $|\cdot|$ denotes the element number of set. (3) Because a frequency leakage arises in 3850 Hz around a large component 3875 Hz and the estimated target number $|\widehat{X}|$ is not large, we put 3850 Hz in dictionary F. So the used F is $\widehat{\mathrm{X}} \cup \{3850\}$; that is, $F = \{3750, 3775, 3850, 3875\}$ in this step, and then the element number of F is $S = |F| = 4$. (4) The sparse scenario $\widehat{\mathbf{s}}$ represented in joint angle-Doppler domain is reconstructed from model (31) as shown in Figure 4. In the figure, the horizontal axis n is the grid index of joint angle-Doppler domain. An arbitrary nth grid in the horizontal axis can be written as $n = (p-1) \cdot S + q$, where $p \in [1, N]$ denotes the pth spatial grid of discretized Ω and $q \in [1, S]$ denotes qth frequency of dictionary F.

As depicted in Figure 4, target 1 occupies two grids in joint angle-Doppler domain; that is, the energy of target 1 spreads into two frequencies in dictionary F because we use a representation on discretized set F_B to approximate an actual Doppler defined in a continuous frequency domain. According to $\widehat{\mathbf{s}}$, the sparse scenario $\widehat{\mathbf{s}}_\Omega$ defined in spatial domain Ω can be obtained from (35) as shown in Figure 5.

In Figure 5, the horizontal axis is the index of discretized azimuth and vertical axis corresponds to discretized elevation in Ω. It is seen that the performance of the proposed strategy is better than traditional MUSIC with less and lower sidelobes around actual targets.

4.2. Resolution of Two Adjacent Targets. With respect the scenario of two adjacent targets in Ω at the same range cell for active radar, the performance of the proposed strategy is investigated in different SNRs. In each experiment, two

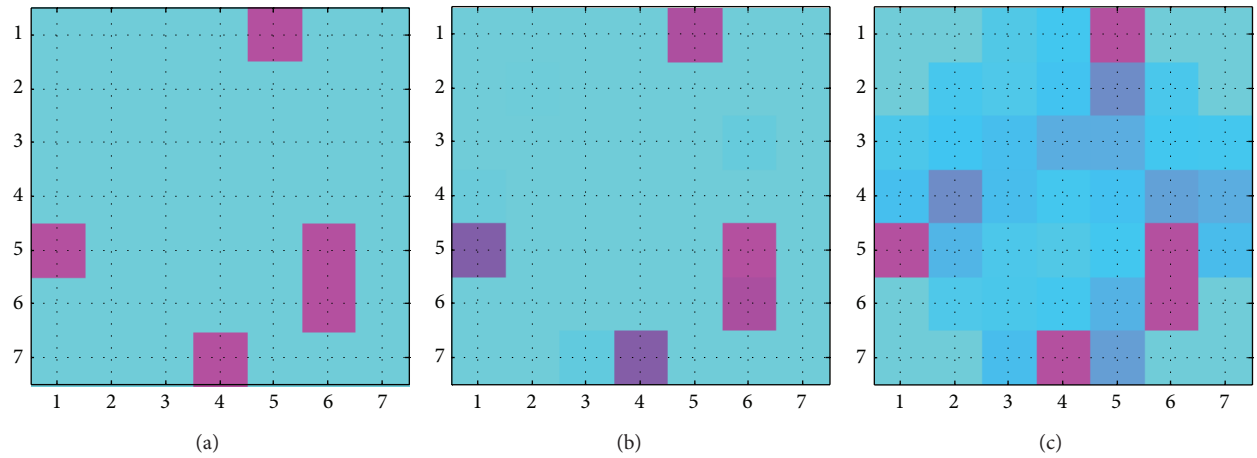


Figure 8: Recovered target scenario with 5 randomly distributed targets. (a) Original Scenario. (b) Recovery of the proposed strategy. (c) Recovery of MUSIC.

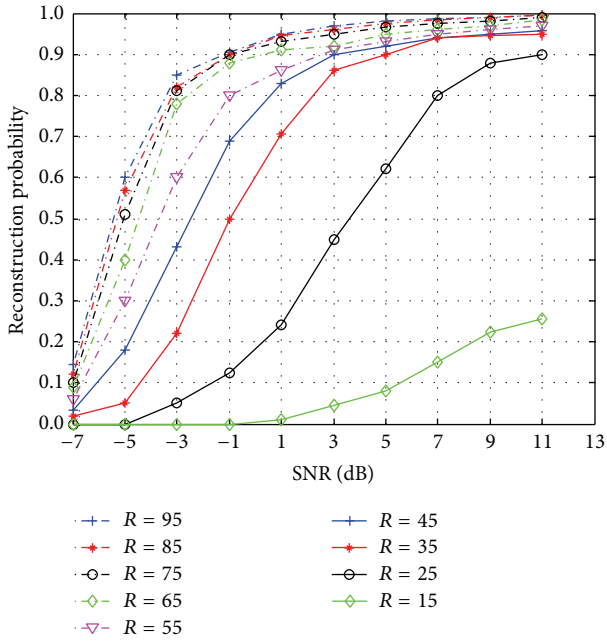


Figure 9: Reconstruction probability of the proposed strategy when $K = 5$.

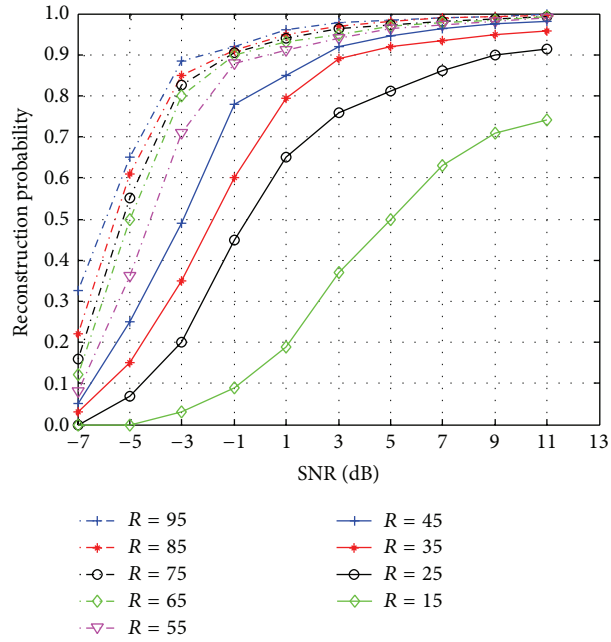


Figure 10: Reconstruction probability of the proposed strategy when $K = 3$.

adjacent angles are randomly selected from Ω. As shown in Figure 6, the reconstruction probability of the proposed strategy is improved with the increase of SNR and is larger than MUSIC algorithm. The proposed strategy achieves more than 90% successful recovery probability when the SNR is larger than −1 dB. According to the result of this subsection, the proposed strategy can determine the target scenarios including two adjacent targets randomly distributed in the spatial domain Ω in the forward-looking direction.

4.3. Resolution of Multiple Randomly-Distributed Targets. In this subsection, the performance of the proposed strategy with respect to the sparse scenario where multiple targets are randomly distributed in Ω is evaluated. In each experiment, the measurement number R in model (31) is equal to 96 and R' in model (40) is 100. As shown in Figure 7, the proposed approach shows better performance than MUSIC, especially at lower SNR level. The performance of the proposed strategy is almost not influenced by the number of targets whereas the performance of MUSIC obviously decays along with the increasing number of targets.

Figure 8 shows the recovered target scenario of one numerical experiment when the experiment point is $\mathrm{St} = \{5, 1\}$. As illustrated in Figure 8, there are less and lower sidelobes around actual targets in the reconstructed scenario of the proposed strategy than MUSIC algorithm.

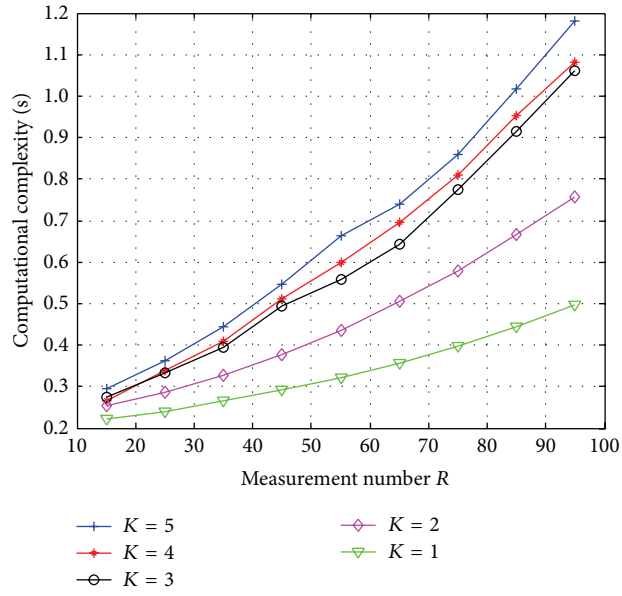


Figure 11: Computational complexity of the proposed strategy.

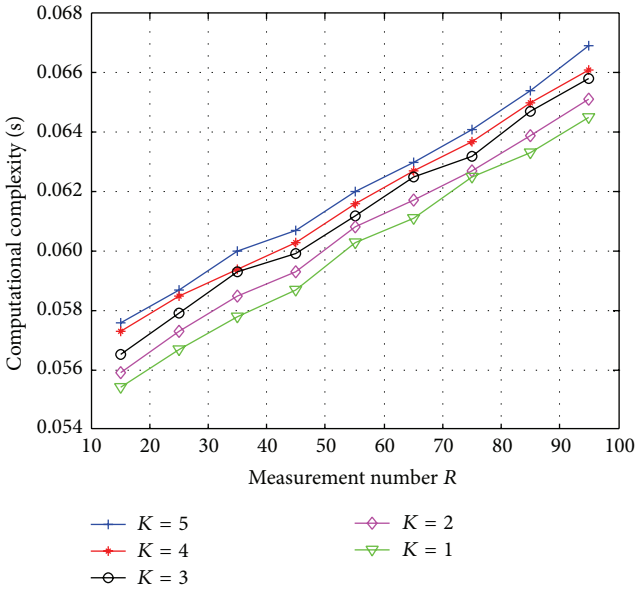


Figure 12: Computational complexity of the MUSIC algorithm.

According to the results in this subsection, the proposed strategy can determine the sparse scenario containing multiple randomly distributed targets in the spatial domain Ω in the forward-looking direction for active radar.

4.4. The Influence of Measurement Number. In the subsection, the reconstruction probability of the proposed strategy along with different measurement number R is investigated. Figure 9 shows the performance of the proposed strategy when target number $K = 5$. As depicted in Figure 9, there is little difference among the reconstructed probabilities when R takes large values, for example, $R \in [65, 95]$, but the difference becomes remarkable when R takes small values, for example, $R \in [15, 35]$. The dramatic increment of the recovery probability is the common characteristic of CS recovery procedure when the measurement number R takes small value. The recovery performance when target number $K = 3$ is depicted in Figure 10. As shown in Figure 10, the recovery probability when $K = 3$ is subject to similar characteristic as $K = 5$.

According to Figures 9 and 10, the recovery probability may be almost changeless when the measurement number R takes the values which are larger than a threshold R_{th}; for example, R_{th} may be equal to 65 when $K = 5$ and may be equal to 55 when $K = 3$. In nature, R_{th} can be seen as the smallest value which meets the condition $R \geq C\mu^2(\mathbf{\Phi}, \mathbf{A}^s)K \log(SN)$, which can assure that the equivalent dictionary $\mathbf{D} = \mathbf{\Phi}\mathbf{A}^s$ satisfies RIP in CS framework. According to Figures 9 and 10, the recovery probability of the proposed strategy can reach a desirable level if the SNR is sufficiently high although the measurement number R take a lower value. For example, the reconstruction probability will be more than 90% when SNR ≥ 3 if the measurement number $R \geq 45$ as shown in Figure 10.

4.5. Computational Complexity. In this subsection, the computational complexity of the proposed approach is addressed through numerical experiments. In the numerical experiments, the measurement number R in model (31) varies from 15 to 95 to evaluate the computational cost with respect to target number $K \in [1, 5]$. The hardware platform to carry out the experiments is "Intel Core 2 Duo CPU E7400 at 2.80 GHz, 3G Memory," and the software is "Matlab R2011a." The computational complexity of the proposed strategy is depicted in Figure 11.

As shown in Figure 11, the computational cost of the proposed strategy becomes more expensive when the target number K and the measurement number R become larger, and the cost slope becomes larger along with the increment of measurement number R and target number K.

As shown in Figure 12, the computational cost of the MUSIC algorithm slightly increases with the measurement number R and the target number K. According to Figures 11 and 12, the computation cost of the proposed strategy is more expensive than the MUSIC algorithm. The difference of the computational complexity between the proposed strategy and MUSIC algorithm becomes more obvious while the measurement number R becomes larger. The running time of the proposed strategy is approximately 17 times more than MUSIC when $R = 95$ and $K = 5$, however, 4 times when $R = 15$ and $K = 1$. Consequently, the computational complexity should be paid more attention when the measurement number R takes large values in the proposed strategy.

5. Conclusions

Based on compressive sensing framework, a strategy using joint angle-Doppler representation basis is proposed which can determine a sparse target scenario in spatial domain

at the same range for active radar in the forward-looking direction. The proposed approach does settle the trouble that traditional SAR and DBS techniques cannot provide an image for active radar in the line of sight and needs only single-receiver channel without any modification on traditional radar hardware. Compared with MUSIC algorithm which needs multiple receiver channels, the proposed strategy shows better performance with different setup about SNR level and target numbers. The improvement of the reconstruction performance when targets do not accurately lie in the discretized spatial grid of Ω is the future work.

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 Dr. Yinan Zhao for proofreading the paper and his valuable suggestions about the recovered procedure in the paper. This work was supported by the National Natural Science Foundation of China under Grant 61371181.

References

[1] E. J. Candes, J. Romberg, and T. Tao, "Robust uncertainty principles: exact signal reconstruction from highly incomplete frequency information," *Institute of Electrical and Electronics Engineers. Transactions on Information Theory*, vol. 52, no. 2, pp. 489–509, 2006.

[2] E. J. Candes and M. B. Wakin, "An introduction to compressive sampling," *IEEE Signal Processing Magazine*, vol. 25, no. 2, pp. 21–30, 2008.

[3] J. H. G. Ender, "On compressive sensing applied to radar," *Signal Processing*, vol. 90, no. 5, pp. 1402–1414, 2010.

[4] I. Bilik, "Spatial compressive sensing for direction-of-arrival estimation of multiple sources using dynamic sensor arrays," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 47, no. 3, pp. 1754–1769, 2011.

[5] S. F. Cotter, "Multiple snapshot matching pursuit for direction of arrival (DOA) estimation," in *Proceedings of the 15th European Signal Processing Conference (EUSIPCO '07)*, pp. 247–251, Poznan, Poland, September 2007.

[6] A. Gretsistas and M. Plumbley, "A multichannel spatial compressed sensing approach for direction of arrival estimation," in *Proceedings of the 9th International Conference on Latent Variable Analysis and Signal Separation (LVA/ICA '10)*, pp. 458–465, St. Malo, France, 2010.

[7] J. Tropp, A. C. Gilbert, and M. J. Strauss, "Simultaneous sparse approximation via greedy pursuit," in *Proceedings of the IEEE International Conference of Acoustics, Speech and Signal Processing*, vol. 5, pp. 721–724, Philadelphia, Pa, USA, March 2005.

[8] E. van den Berg and M. P. Friedlander, "Theoretical and empirical results for recovery from multiple measurements," *IEEE Transaction on Information Theory*, vol. 56, no. 5, pp. 2516–2527, 2010.

[9] S. Ji, Y. Xue, and L. Carin, "Bayesian compressive sensing," *IEEE Transactions on Signal Processing*, vol. 56, no. 6, pp. 2346–2356, 2008.

[10] S. Ji, D. Dunson, and L. Carin, "Multitask compressive sensing," *IEEE Transactions on Signal Processing*, vol. 57, no. 1, pp. 92–106, 2009.

[11] G. Tzagkarakis, D. Milioris, and P. Tsakalides, "Multiple-measurement Bayesian compressed sensing using GSM priors for DOA estimation," in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP '10)*, pp. 2610–2613, Dallas, Tex, USA, March 2010.

[12] L. Gan, X. Q. Wang, and H. S. Liao, "DOA estimation of coherently distributed sources based on block-sparse constraint," *IEICE Transactions on Communications*, vol. 95, no. 7, pp. 2472–2476, 2012.

[13] M. Carlin, P. Rocca, G. Oliveri, F. Viani, and A. Massa, "Directions-of-arrival estimation through Bayesian compressive sensing strategies," *IEEE Transactions on Antennas and Propagation*, vol. 61, no. 7, pp. 3828–3838, 2013.

[14] J. M. Kim, O. K. Lee, and J. C. Ye, "Compressive MUSIC: revisiting the link between compressive sensing and array signal processing," *IEEE Transactions on Information Theory*, vol. 58, no. 1, pp. 278–301, 2012.

[15] X. Zhan, R. Zhang, D. Yin, and C. Huo, "SAR image compression using multiscale dictionary learning and sparse representation," *IEEE Geoscience and Remote Sensing Letters*, vol. 10, no. 5, pp. 1090–1094, 2013.

[16] X. X. Zhu and R. Bamler, "Tomographic SAR inversion by L1-norm regularization-the compressive sensing approach," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 48, no. 10, pp. 3839–3846, 2010.

[17] E. J. Candès, Y. C. Eldar, D. Needell, and P. Randall, "Compressed sensing with coherent and redundant dictionaries," *Applied and Computational Harmonic Analysis*, vol. 31, no. 1, pp. 59–73, 2011.

[18] CVX Research Inc., "cvx toolbox," 2014, http://cvxr.com/.

Hindawi Publishing Corporation
Mathematical Problems in Engineering
Volume 2014, Article ID 432529, 7 pages
http://dx.doi.org/10.1155/2014/432529

Research Article

A Study on Bottom Friction Coefficient in the Bohai, Yellow, and East China Sea

Daosheng Wang,[1] Qiang Liu,[2] and Xianqing Lv[1]

[1] *Laboratory of Physical Oceanography, Ocean University of China, Qingdao 266100, China*
[2] *College of Engineering, Ocean University of China, Qingdao 266100, China*

Correspondence should be addressed to Qiang Liu; liuqiang@ouc.edu.cn

Received 9 April 2014; Revised 10 June 2014; Accepted 12 June 2014; Published 1 July 2014

Academic Editor: Fatih Yaman

The adjoint tidal model based on the theory of inverse problem has been applied to investigate the effect of bottom friction coefficient (BFC) on the tidal simulation. Using different schemes of BFC containing the constant, different constant in different subdomain, depth-dependent form, and spatial distribution obtained from data assimilation, the M_2 constituent in the Bohai, Yellow, and East China Sea (BYECS) is simulated by assimilating TOPEX/Poseidon altimeter data, respectively. The simulated result with spatially varying BFC obtained from data assimilation is better than others. Results and analysis of BFC in BYECS indicate that spatially varying BFC obtained from data assimilation is the best fitted one; meanwhile it could improve the accuracy in the simulation of M_2 constituent. Through the analysis of the best fitted one, new empirical formulas of BFC in BYECS are developed with which the commendable simulated results of M_2 constituent in BYECS are obtained.

1. Introduction

The bottom friction plays a significant role in the tidal phenomenon. In numerical simulations of tide, bottom friction is generally parameterized by the bottom friction coefficient (BFC). In order to improve the simulation accuracy, it is essential to determine the BFC correctly. In previous studies [1–8], several methods were suggested to determine the BFC and some encouraging simulated results were achieved. Lee and Jung [9] used a three-dimensional mode-splitting, σ-coordinate barotropic finite-difference model to examine M_2 tidal elevation and current in the Yellow Sea and East China Sea, and they treated the BFC as a constant in the whole computing domain. Zhao et al. [10] simulated the semidiurnal and diurnal tides and tidal currents in the whole Eastern China Seas with different BFC in different subdomain. Kang et al. [11] carried out a fine grid tidal modeling experiment to study the tidal phenomena in the Yellow and East China Seas, and they used the depth-dependent form of BFC. He et al. [12] set up a numerical adjoint model with TOPEX/Poseidon (T/P) altimeter data to investigate the shallow water tidal constituents in the Bohai and Yellow Sea. In their model, the Bohai and Yellow Sea were divided into five sub-regions with different BFC. Lu and Zhang [13] used the adjoint method to assimilate T/P altimeter data into a 2-dimensional tidal model in the Bohai, Yellow, and East China Sea (BYECS) and the spatially varying BFC were estimated with the independent point strategy.

Additionally, open boundary conditions (OBCs) are crucial for the representation of tidal processes in the regional ocean model [14]. Generally, OBCs could be obtained from the larger scale model or by interpolating the existing observation data near the location. However, OBCs obtained by the methods mentioned above have to be adjusted by experience to get ideal simulated results. Based on the theory of inverse problem, the adjoint method is a powerful tool for parameter estimation [15], and thus OBCs could be optimized automatically. Zhang and Lu [16] applied the four-dimensional variational data assimilation technology to simulate the three-dimensional tidal currents in the marginal seas and the OBCs were optimized. Guo et al. [14] estimated the OBCs in Bohai Sea by an adjoint data assimilation approach with independent point strategy and obtained good simulated result of M_2 constituent. Zhang and Wang [17] developed

a new method based on the adjoint method to inverse the periodic OBCs in two-dimensional tidal models and used it to simulate the M_2 constituent in BYECS successfully.

As mentioned above, BFC is an important parameter for tidal models and many schemes of BFC have been used in previous study. However, so far there are few systematic comparisons about the different schemes of BFC. Because different numerical models and observations are used in different studies, the simulated results in those papers in which the BFC are different could not be compared directly. In this paper, firstly the adjoint tidal model is employed to compare some different schemes of BFC. At the same time, in order to reduce the influence of OBCs that are also important for tidal models, we use the adjoint method to optimize OBCs. Based on the simulation of M_2 constituent in BYECS, several different schemes of BFC including the constant, different constant in different subdomain, depth-dependent form, and spatial distribution obtained from data assimilation are compared to find the best fitted one. Then we try to analyze the best fitted one to set up new empirical formulas of BFC in BYECS with which the preferable simulated results could be obtained.

2. Adjoint Tidal Model

2.1. Equations. The governing equations are described under the rectangular coordinate system. Assuming that pressure is hydrostatic and density is constant, the depth averaged two-dimensional tidal model is as follows:

$$\begin{aligned}
&\frac{\partial \zeta}{\partial t} + \frac{\partial \left[(h+\zeta)u\right]}{\partial x} + \frac{\partial \left[(h+\zeta)v\right]}{\partial y} = 0,\\
&\frac{\partial u}{\partial t} + u\frac{\partial u}{\partial x} + v\frac{\partial u}{\partial y} - fv + \frac{ku\sqrt{u^2+v^2}}{h+\zeta}\\
&\qquad - A\left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2}\right) + g\frac{\partial \zeta}{\partial x} = 0,\\
&\frac{\partial v}{\partial t} + u\frac{\partial v}{\partial x} + v\frac{\partial v}{\partial y} + fu + \frac{kv\sqrt{u^2+v^2}}{h+\zeta}\\
&\qquad - A\left(\frac{\partial^2 v}{\partial x^2} + \frac{\partial^2 v}{\partial y^2}\right) + g\frac{\partial \zeta}{\partial y} = 0,
\end{aligned} \tag{1}$$

where t is time, x and y are Cartesian coordinates, h is undisturbed water depth, ζ is sea surface elevation above the undisturbed sea level, u and v are velocity components in the east and north, f is the Coriolis parameter, g is the acceleration due to gravity, k is the BFC, and A is the horizontal eddy viscosity coefficient.

With the adjoint method described in Lu and Zhang [13], the cost function is constructed as

$$J = \frac{1}{2}K_\zeta \int_\Sigma \left(\zeta - \widehat{\zeta}\right)^2 d\sigma, \tag{2}$$

where K_ζ is a constant and Σ is the set of the observation locations.

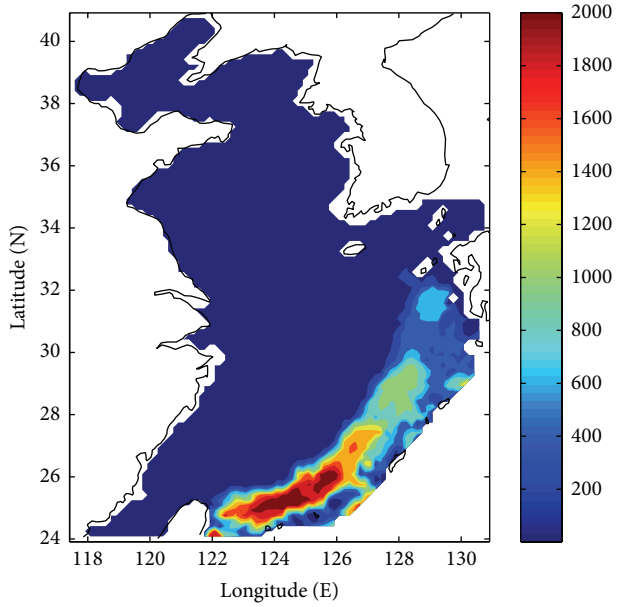


Figure 1: Bathymetry map of BYECS.

And the adjoint model can be constructed as follows:

$$\begin{aligned}
&\frac{\partial \lambda}{\partial t} + u\frac{\partial \lambda}{\partial x} + v\frac{\partial \lambda}{\partial y} + \frac{k\mu u\sqrt{u^2+v^2}}{(h+\zeta)^2} + \frac{kvv\sqrt{u^2+v^2}}{(h+\zeta)^2}\\
&\qquad + g\frac{\partial \mu}{\partial x} + g\frac{\partial v}{\partial y} = K_\zeta\left(\zeta - \widehat{\zeta}\right),\\
&\frac{\partial \mu}{\partial t} - \left(f + \frac{kuv}{(h+\zeta)\sqrt{u^2+v^2}}\right)v - \mu\frac{\partial u}{\partial x} - v\frac{\partial v}{\partial x}\\
&\qquad + \frac{\partial}{\partial x}(\mu u) + \frac{\partial}{\partial y}(\mu v) + (h+\zeta)\frac{\partial \lambda}{\partial x}\\
&\qquad + A\left(\frac{\partial^2 \mu}{\partial x^2} + \frac{\partial^2 \mu}{\partial y^2}\right) - \frac{k\left(2u^2+v^2\right)}{(h+\zeta)\sqrt{u^2+v^2}}\mu = 0,\\
&\frac{\partial v}{\partial t} + \left(f - \frac{kuv}{(h+\zeta)\sqrt{u^2+v^2}}\right)\mu - \mu\frac{\partial u}{\partial y}\\
&\qquad - v\frac{\partial v}{\partial y} + \frac{\partial}{\partial x}(vu) + \frac{\partial}{\partial y}(vv) + (h+\zeta)\frac{\partial \lambda}{\partial y}\\
&\qquad + A\left(\frac{\partial^2 v}{\partial x^2} + \frac{\partial^2 v}{\partial y^2}\right) - \frac{k\left(u^2+2v^2\right)}{(h+\zeta)\sqrt{u^2+v^2}}v = 0,
\end{aligned} \tag{3}$$

where ζ is the simulated result, $\widehat{\zeta}$ is the observation, and λ, μ, and v denote the adjoint variables of ζ, u, and v, respectively.

The finite difference schemes of (1) and (3) are similar to those in Lu and Zhang [13].

2.2. Model Setting. The computing area is BYECS (117.5°E–131°E, 24°N–41°N) which is shown in Figure 1. The horizontal resolution is $10' \times 10'$. The time step is 62.103

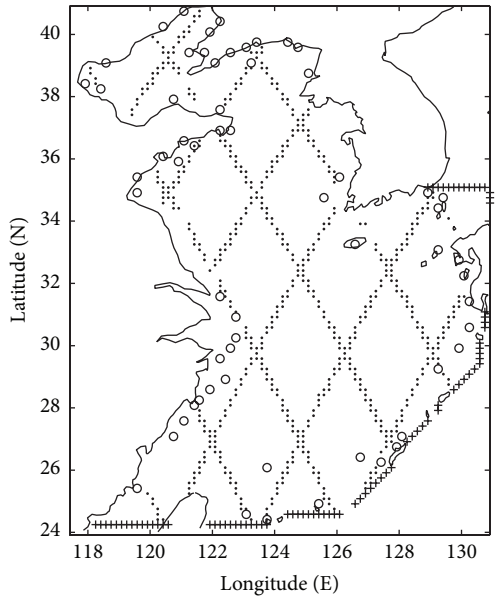


FIGURE 2: Positions of T/P altimeter tracks ("•") and tidal gauges ("o") and open boundary ("+").

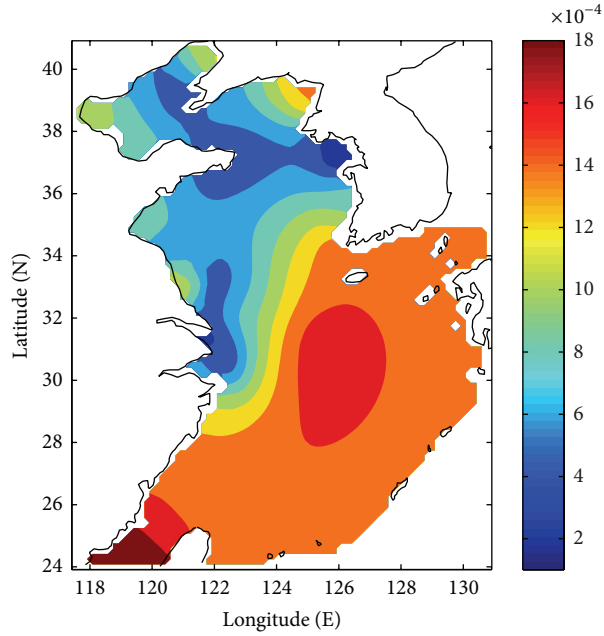


FIGURE 3: The BFC distribution in E4.

seconds, which is 1/720 of the period of M_2 constituent. The eddy viscosity coefficient (A) is 5000 m^2/s. The positions of tidal gauge stations, the T/P altimeter tracks, and the open boundary are shown in Figure 2.

3. Numerical Experiments and Result Analysis

3.1. Calculation Process of Numerical Experiments. Initial conditions are that the sea surface elevation (ζ) and the velocities (u and v) are zero. In addition, the initial values of OBCs are set to zero.

The calculation process of the adjoint tidal model is designed as follows.

(1) With the BFC given, which is fixed in the whole computing process, OBCs existed and other model parameters run the forward tidal model.

(2) The difference of water elevation between simulated results from step (1) and observations at the grid points on T/P satellite tracks serves as the external force of the adjoint model. Values of adjoint variables are obtained through backward integration of the adjoint equations.

(3) With the values of adjoint variables from the adjoint model, the OBCs could be adjusted by the method mentioned in Cao et al. [18].

Repeat steps (1)–(3) until the number of iteration steps is exactly 100. For the setting of adjoint tidal model in this study, 100 iteration steps are sufficient because both the cost function and the difference between observations and simulated results will decrease slowly after this step.

3.2. Setting of Numerical Experiments. In each numerical experiment, the BFC is fixed and the OBCs are optimized by assimilating T/P altimeter data into the adjoint tidal model, so that we could compare the different schemes of BFC adequately without the possibility that the OBCs do not match the BFC. Moreover, the tide gauge data are used as an independent check of the model fidelity.

Refer to some schemes of BFC generally used in previous studies, and we design several numerical experiments to compare them.

E1: the BFC is treated as a constant (0.0015) in BYECS.

E2: the BFC is depth-dependent form which is similar to that used by Kang et al. [11]. The BFC is defined by $k = g/C^2$, where g is gravity acceleration, C is Chezy coefficient, and the depth-dependent form of the Chezy coefficient are applied as $C = h^{1/6}/n$ with $n = 0.023$.

E3: the scheme of BFC is the same as that employed in Zhao et al. [10]. The BFC is taken to be 0.001 at the west of the line from (25°15′N, 120°45′E) to (40°00′N, 124°15′E), 0.0035 in the Korean Strait, and 0.0016 in other areas.

E4: the space-varying BFC is obtained by assimilating observations using the adjoint method in Lu and Zhang [13]. The difference is that the initial condition of BFC in this paper is 0.0015. The spatial distribution of BFC is shown in Figure 3.

3.3. Results of Numerical Experiments. When the tide is stable, the results of next period are used to do harmonic analysis. The mean absolute errors (MAEs) in amplitude and phase between simulation results and observations (T/P data and tidal gauge data) are shown in Table 1.

Table 1: Differences between simulated results and observations (T/P data and tidal gauge data).

EXP	MAEs of T/P data		MAEs of tidal gauge data	
	Amplitude (cm)	Phase lag (°)	Amplitude (cm)	Phase lag (°)
E1	7.2	6.2	10.2	7.3
E2	7.6	6.7	10.3	8.9
E3	6.9	6.1	9.9	7.2
E4	5.7	5.8	6.7	6.6

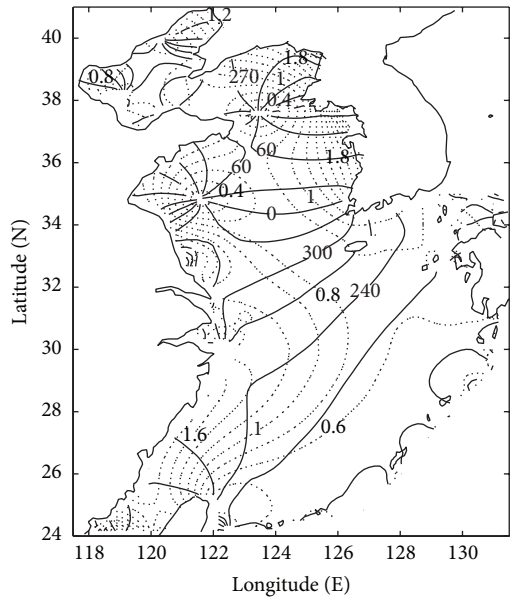


Figure 4: The cotidal chart obtained from E4 (the dashed line denotes coamplitude line (m), and solid line denotes cophase line (degree)).

From Table 1, one can find that E4 obtains the best simulated result. From the MAEs in amplitude and phase between simulation and T/P data, it could be found that E4 obtains the best assimilated results in the same steps of assimilation. And it is obvious that MAEs between simulation and tidal gauge data are minimum. We try to increase the number of iteration steps in E1, E2, and E3, but no improvements are achieved.

The cotidal chart of M_2 constituent obtained in E4 is shown in Figure 4. Compared with Lefèvre et al. [19] and Fang et al. [20], the cotidal chart seems to coincide with the observed M_2 constituent in BYECS fairly well. It also proves that E4 gets perfect simulated result. As shown in Figure 4, there are two amphidromic points in the Bohai Sea, one of which is near Qinhuangdao and the other is near the Yellow River delta. There are also two amphidromic points in the Yellow Sea, one of which is north of Chengshantou and the other is southeast of Qingdao.

4. Discussion of BFC

4.1. Discussion from Numerical Results. As shown by Table 1 and Figure 4, it is obvious that E4 with the space-varying BFC obtains the best simulated result.

Mofjeld [21] used a turbulence closure model to investigate the dependence on water depth of bottom stress and quadratic drag coefficient for a steady barotropic pressure-driven current in unstratified water when the current was the primary source of turbulence. He noted that the quadratic drag coefficient was approximated reasonably well by a formula from nonrotating channel theory in which the coefficient depended only on the ratio of the water depth to the bottom roughness. Jenter and Madsen [22] studied the bottom stress in wind-stress depth-average coastal flows and found that the drag tensor variation was a function of water depth, wind stress, and bottom roughness. From the aforementioned studies, it is seen that the BFC generally depends on the water depth and bottom roughness. And there is no doubt that the water depth and bottom roughness are diverse in different area and they vary spatially. In addition, Kagan et al. [23] studied the impact of the spatial variability in bottom roughness on tidal dynamics and energetics in the North European Basin and indicated that ignoring the spatial variability in bottom roughness was only partially correct because it was liable to break down for the tidal energetics. Therefore, the BFC should be spatially varying in fact. It is noticeable that the schemes of a constant BFC like in E1 is not reasonable enough. The space-varying BFC obtained from the data assimilation seems to be more advisable in physics.

In fact, BFC in E2 is depth-dependent, and thus it is also spatially varying. However, the simulated results from E2 are worse than that from E1 and E3 and much worse than that from E4. From Figure 3, the BFC in shallow water are larger than those in deep water in the Bohai Sea and the Yellow Sea individually. Meanwhile, the average water depth of the Bohai Sea is 19.3 m and the average BFC is 0.00082, while they were 45.4 m and 0.00081 for the Yellow Sea and 334.7 m and 0.0015 for the East China Sea. From the definition of BFC in E2, it is evident that the BFC and the depth are in inverse proportion in whole region. In detail, the average BFC of the Bohai Sea is 0.0021, while it is 0.0017 for the Yellow Sea and 0.0011 for the East China Sea. We can find that the BFC in the Bohai Sea and the Yellow Sea has the same changing trend with E4, but the value is larger. Green and McCave [24] indicated that the form drag caused by the bottom topography, wave-current interaction, boundary-layer stratification, and so on may impact the BFC. The water depth changes largely in the Okinawa trough, so the form drag should be larger. But in E2 the BFC in East China Sea is small and the East China Sea is the largest area in BYECS, so the simulated result of E2 is dissatisfactory. We surmise that the scheme of BFC in E2 may be reasonable in the shelf sea and not applicable in the area of slope and trough. Thus, it can be seen that the spatially varying BFC from data assimilation is better than the depth-dependent form in BYECS, especially in the East China Sea.

In addition, E3 obtains better result than E2. In E2, the BFC is 0.0019 at the west of the line in E3, 0.0012 in the Korean Strait, and 0.0014 in other areas; meanwhile they are 0.0008,

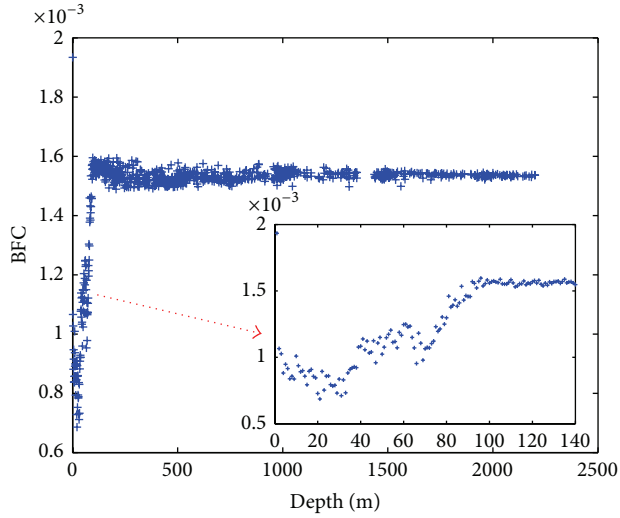


Figure 5: BFC versus water depth.

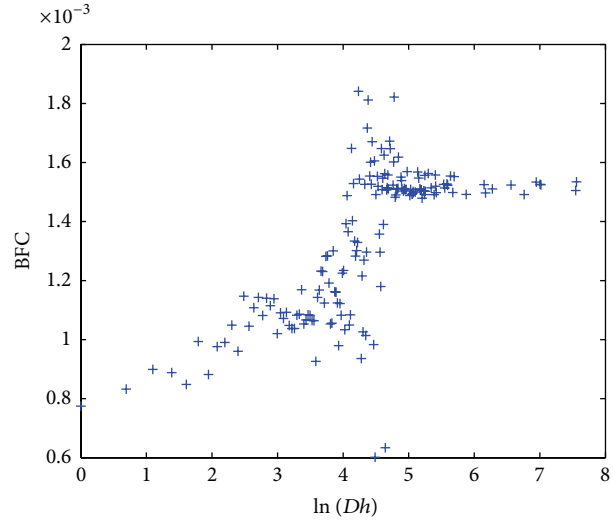


Figure 6: BFC versus ln(Dh) when water depth is less than 100 meters.

0.0014, and 0.0015 in E4. In the areas except the Korean Strait whose area is small, the BFC in E3 and E4 have the same changing tendency and the average values are approximately equal. However there is the opposite trend in E2. It seems to explain that the BFC in E3 is better than that in E2. And it proves that the scheme of BFC in E4 is the best fitted one from another side.

In conclusion, the spatially varying BFC in E4 is the best fitted BFC in BYECS.

4.2. Further Exploration of BFC. In this section, the scheme of BFC in E4 is analyzed to investigate the relationship between BFC and water depth, the change rate of seafloor topography (CRST), and bottom roughness.

In this study, CSRT is described as follows:

$$Dh = \left|h_{i,j} - h_{i+1,j}\right| + \left|h_{i,j} - h_{i-1,j}\right| + \left|h_{i,j} - h_{i,j+1}\right| + \left|h_{i,j} - h_{i,j-1}\right|. \quad (4)$$

And bottom roughness is described as follows:

$$DDh = \left(h_{i,j} - h_{i+1,j}\right)^2 + \left(h_{i,j} - h_{i-1,j}\right)^2 + \left(h_{i,j} - h_{i,j+1}\right)^2 + \left(h_{i,j} - h_{i,j-1}\right)^2. \quad (5)$$

The correlation coefficient between BFC and water depth is 0.4540, while it is 0.3845 for CRST and 0.2520 for bottom roughness. It is shown that water depth is the significant factor that affects BFC. We demonstrate the BFC versus water depth in Figure 5 and could find that the BFC is a constant when water depth is larger than 100 meters. However, when water depth is less than 100 meters, the BFC varies complicatedly. So we focus on the study of quantitative relations of BFC with water depth, CRST and bottom roughness when water depth is less than 100 meters.

Considering BFC is mainly affected by water depth, in first step we ignore the CSRT and bottom roughness to make the relationship simple and just investigate the quantitative relationship of BFC and water depth. From partial enlarged drawing in Figure 5, it could be seen that there are two sections. When water depth is less than 30 meters, BFC decreases with the water depth increasing, while increasing for larger than 30 meters. The fitting function could be obtained as follows:

$$k = \begin{cases} 1.5363 \times 10^{-3}, & h \geq 100 \\ (56.8850 + 0.9674h) \times 10^{-5}, & 30 \leq h < 100 \\ (100.0 - 0.5413h) \times 10^{-5}, & h < 30. \end{cases} \quad (6)$$

From another perspective, a linear function could describe the relationship between BFC and water depth roughly, and at the same time Dh and DDh are also considered. As seen in Figure 6, BFC increases linearly along with ln(Dh) by and large. From Figure 7, it is shown that it is difficult to use a formula to describe the relationship between BFC and ln(DDh). Therefore considering the impact of h, Dh upon BFC, we obtain the formula as follows:

$$k = \begin{cases} 1.5363 \times 10^{-3}, & h \geq 100 \\ (0.5255 + 0.0068h + 0.0731 \ln(Dh)) \times 10^{-3}, & h < 100. \end{cases} \quad (7)$$

Using formulas (6) and (7), two new schemes of BFC in BYECS are obtained, and they are recorded as E5 and E6. The differences between simulated results and observations are shown in Table 2.

From Tables 1 and 2, it could be found that the simulated results of E5 and E6 are better than those of others except E4. It indicates that the schemes of BFC obtained from the statistical relation could describe the BFC in BYECS preferably and improve the result of numerical simulation.

Through the analysis of the scheme of BFC in E4, we set up new empirical formulas of BFC in BYECS with which the commendable simulated results are obtained. It should

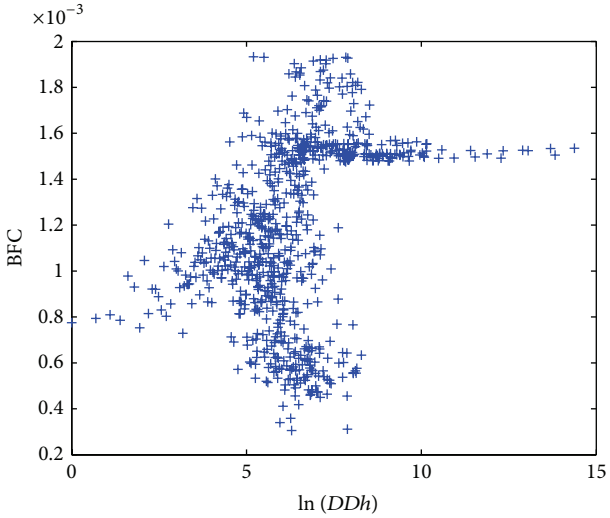


Figure 7: BFC versus ln(*DDh*) when water depth is less than 100 meters.

Table 2: Differences between simulated results and observations (T/P data and tidal gauge data).

EXP	MAEs of T/P data		MAEs of tidal gauge data	
	Amplitude (cm)	Phase lag (°)	Amplitude (cm)	Phase lag (°)
E5	6.5	5.9	8.5	6.5
E6	6.7	6.0	8.4	6.1

be noted that the calculation of BFC in BYECS by the new empirical formulas just needs the bathymetric data. So it can be considered to be referenced in the simulation of M_2 constituent in BYECS.

5. Conclusions

The adjoint tidal model based on the theory of inverse problem has been applied to investigate the effect of BFC on the tidal simulation. The M_2 constituent in BYECS is simulated by assimilating T/P altimeter data with several different schemes of BFC: the constant, different constant in different subdomain, depth-dependent form, and spatial distribution obtained from data assimilation. Comparing with the observations at tidal gauges, it is found that the simulated result with the spatially varying BFC is the best, and the MAEs in amplitude and phase are 6.7 cm and 6.6°, respectively, while the least values in other experiments are 9.9 cm and 7.2°. Comparing with the observations at T/P stations, we found that the simulated result with spatially varying BFC has advantages over others and the MAEs in amplitude and phase are 5.7 cm and 5.8°, respectively, while in other experiments they are at least 6.9 cm and 6.1°. The simulated results and the analysis of BFC in BYECS simultaneously indicate that spatially varying BFC obtained from data assimilation is the best fitted one, and it could improve the accuracy in the simulation of M_2 constituent. Finally, through the statistical analysis of the spatially varying BFC obtained from data assimilation, new empirical formulas of BFC in BYECS are obtained. We found that the simulated results with new empirical formulas are better than traditional schemes, such as the constant, different constant in different subdomain, and depth-dependent form. We believe that the new empirical formulas could be referenced in the simulation of M_2 constituent in BYECS.

Conflict of Interests

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

Acknowledgments

The authors deeply thank the reviewers and editor for their constructive criticism of an early version of the paper. Partial support for this research was provided by the National Natural Science Foundation of China through Grants nos. 41072176 and 41371496, the National Science and Technology Support Program through Grant no. 2013BAK05B04, the State Ministry of Science and Technology of China through Grant no. 2013AA122803, and the Fundamental Research Funds for the Central Universities 201362033 and 201262007.

References

[1] C. Chen, H. Huang, R. C. Beardsley, H. Liu, Q. Xu, and G. Cowles, "A finite volume numerical approach for coastal ocean circulation studies: comparisons with finite difference models," *Journal of Geophysical Research C: Oceans*, vol. 112, no. 3, Article ID C03018, 2007.

[2] L. S. Quaresma and A. Pichon, "Modelling the barotropic tide along the West-Iberian margin," *Journal of Marine Systems*, vol. 109-110, pp. S3–S25, 2013.

[3] J. Zhang and X. Lu, "Parameter estimation for a three-dimensional numerical barotropic tidal model with adjoint method," *International Journal for Numerical Methods in Fluids*, vol. 57, no. 1, pp. 47–92, 2008.

[4] G. D. Egbert, R. D. Ray, and B. G. Bills, "Numerical modeling of the global semidiurnal tide in the present day and in the last glacial maximum," *Journal of Geophysical Research C: Oceans*, vol. 109, no. 3, Article ID C03003, 2004.

[5] H. J. Lee, K. T. Jung, J. K. So, and J. Y. Chung, "A three-dimensional mixed finite-difference Galerkin function model for the oceanic circulation in the Yellow Sea and the East China Sea in the presence of M_2 tide," *Continental Shelf Research*, vol. 22, no. 1, pp. 67–91, 2002.

[6] G. Sannino, A. Bargagli, and V. Artale, "Numerical modeling of the semidiurnal tidal exchange through the strait of Gibraltar," *Journal of Geophysical Research C: Oceans*, vol. 109, no. 5, 2004.

[7] A. W. Heemink, E. E. A. Mouthaan, M. R. T. Roest, E. A. H. Vollebregt, K. B. Robaczewska, and M. Verlaan, "Inverse 3D shallow water flow modelling of the continental shelf," *Continental Shelf Research*, vol. 22, no. 3, pp. 465–484, 2002.

[8] M. U. Altaf, M. Verlaan, and A. W. Heemink, "Efficient identification of uncertain parameters in a large-scale tidal model of the European continental shelf by proper orthogonal decomposition," *International Journal for Numerical Methods in Fluids*, vol. 68, no. 4, pp. 422–450, 2012.

[9] J. C. Lee and K. T. Jung, "Application of eddy viscosity closure models for the M_2 tide and tidal currents in the Yellow Sea and the East China Sea," *Continental Shelf Research*, vol. 19, no. 4, pp. 445–475, 1999.

[10] B. Zhao, G. Fang, and D. Cao, "Numerical modeling on the tides and tidal currents in the eastern China Seas," *Yellow Sea Research*, vol. 5, pp. 41–61, 1993.

[11] S. K. Kang, S. Lee, and H. Lie, "Fine grid tidal modeling of the Yellow and East China Seas," *Continental Shelf Research*, vol. 18, no. 7, pp. 739–772, 1998.

[12] Y. He, X. Lu, Z. Qiu, and J. Zhao, "Shallow water tidal constituents in the Bohai Sea and the Yellow Sea from a numerical adjoint model with TOPEX/POSEIDON altimeter data," *Continental Shelf Research*, vol. 24, no. 13-14, pp. 1521–1529, 2004.

[13] X. Lu and J. Zhang, "Numerical study on spatially varying bottom friction coefficient of a 2D tidal model with adjoint method," *Continental Shelf Research*, vol. 26, no. 16, pp. 1905–1923, 2006.

[14] Z. Guo, A. Cao, and X. Lv, "Inverse estimation of open boundary conditions in the Bohai Sea," *Mathematical Problems in Engineering*, vol. 2012, Article ID 628061, 9 pages, 2012.

[15] J. Zhang and H. Chen, "Semi-idealized study on estimation of partly and fully space varying open boundary conditions for tidal models," *Abstract and Applied Analysis*, vol. 2013, Article ID 282593, 14 pages, 2013.

[16] J. Zhang and X. Lu, "Inversion of three-dimensional tidal currents in marginal seas by assimilating satellite altimetry," *Computer Methods in Applied Mechanics and Engineering*, vol. 199, no. 49–52, pp. 3125–3136, 2010.

[17] J. Zhang and Y. Wang, "A method for inversion of periodic open boundary conditions in two-dimensional tidal models," *Computer Methods in Applied Mechanics and Engineering*, vol. 275, pp. 20–38, 2014.

[18] A. Cao, Z. Guo, and X. Lü, "Inversion of two-dimensional tidal open boundary conditions of M_2 constituent in the Bohai and Yellow Seas," *Chinese Journal of Oceanology and Limnology*, vol. 30, no. 5, pp. 868–875, 2012.

[19] F. Lefèvre, C. le Provost, and F. H. Lyard, "How can we improve a global ocean tide model at a regional scale? a test on the Yellow Sea and the East China Sea," *Journal of Geophysical Research C: Oceans*, vol. 105, no. 4, pp. 8707–8725, 2000.

[20] G. Fang, Y. Wang, Z. Wei, B. H. Choi, X. Wang, and J. Wang, "Empirical cotidal charts of the Bohai, Yellow, and East China Seas from 10 years of TOPEX/Poseidon altimetry," *Journal of Geophysical Research C: Oceans*, vol. 109, no. 11, Article ID C11006, 2004.

[21] H. O. Mofjeld, "Depth dependence of bottom stress and quadratic drag coefficient for barotropic pressure-driven currents," *Journal of Physical Oceanography*, vol. 18, pp. 1658–1669, 1988.

[22] H. L. Jenter and O. S. Madsen, "Bottom stress in wind-driven depth-averaged coastal flows," *Journal of Physical Oceanography*, vol. 19, pp. 962–974, 1989.

[23] B. A. Kagan, E. V. Sofina, and E. Rashidi, "Inversion of two-dimensional tidal open boundary conditions of M_2 constituent in the Bohai and Yellow Seas," *Ocean Dynamics*, vol. 62, no. 10–12, pp. 1425–1442, 2012.

[24] M. O. Green and I. N. McCave, "Seabed drag coefficient under tidal currents in the eastern Irish Sea," *Journal of Geophysical Research*, vol. 100, no. 8, pp. 16057–16069, 1995.

Hindawi Publishing Corporation
Mathematical Problems in Engineering
Volume 2014, Article ID 280738, 16 pages
http://dx.doi.org/10.1155/2014/280738

Review Article

A Review on Migration Methods in B-Scan Ground Penetrating Radar Imaging

Caner Özdemir, Şevket Demirci, Enes Yiğit, and Betül Yilmaz

Department of Electrical and Electronics Engineering, Mersin University, Ciftlikkoy, 33343 Mersin, Turkey

Correspondence should be addressed to Caner Özdemir; cozdemir@mersin.edu.tr

Received 14 March 2014; Accepted 5 May 2014; Published 10 June 2014

Academic Editor: Fatih Yaman

Even though ground penetrating radar has been well studied and applied by many researchers for the last couple of decades, the focusing problem in the measured GPR images is still a challenging task. Although there are many methods offered by different scientists, there is not any complete migration/focusing method that works perfectly for all scenarios. This paper reviews the popular migration methods of the B-scan GPR imaging that have been widely accepted and applied by various researchers. The brief formulation and the algorithm steps for the hyperbolic summation, the Kirchhoff migration, the back-projection focusing, the phase-shift migration, and the ω-k migration are presented. The main aim of the paper is to evaluate and compare the migration algorithms over different focusing methods such that the reader can decide which algorithm to use for a particular application of GPR. Both the simulated and the measured examples that are used for the performance comparison of the presented algorithms are provided. Other emerging migration methods are also pointed out.

1. Introduction

Ground penetrating radar (GPR) is remote sensing technique that can nondestructively sense and detect objects inside the visually opaque environment or underneath ground surface. GPR has gained its fame thanks to its high resolution capability and applicability in numerous fields such as detecting mines and unexploded ordinances, finding water leakages, investigating archeological substances, spotting asphalt/concrete cracks in highways, searching buried victims after an earthquake or an avalanche, and imaging behind the wall for security applications [1–4]. Depending on the application, different scanning schemes, namely, A-scan, B-scan, and C-scan, are being employed [1]. In the B-scan measurement situation, a downward looking GPR antenna is moved along a straight path on the top of the surface while the GPR sensor is collecting and recording the scattered field at different spatial positions. This static measured data collected at single point is called an A-scan [1].

The data collection in a typical GPR operation can be either employed in the time domain by recording the scattered response of a time-domain pulse or in the frequency domain by recording the frequency response of the scattered field. For the former case, the two-dimensional (2D) space-time GPR image $I(x, t)$ is attained by conveying a time-domain pulse towards the surface at consecutive, distinct synthetic aperture points. For the latter case, an inverse Fourier transform (IFT) operation should be accommodated to carry the collected data from the spatial-frequency domain to space-time GPR image. In any case, the depth resolution is achieved by the transmitted signal's frequency diversity. The resolution along the scanning direction is attained by the synthetic processing of the received data collected at different spatial points of the B-scan. While a fine resolution in the depth axis is usually easy to get by utilizing a wide-band transmitted signal, the resolution along the scanning direction is much harder to realize and requires special treatment. On the other hand, some specific GPR applications like preservation tasks in ancient constructions or stone masonry structures and detection of antipersonnel landmines for humanitarian demining require high resolution images of the investigated structures or regions especially in the cross-range dimensions.

In a typical space-time B-scan GPR image, any scatterer within the image region shows up as a hyperbola because of

the different trip times of the EM wave while the antenna is moving along the scanning direction. For this reason, the resolution along the synthetic aperture direction depicts undesired low resolution features owing to the long tails of the hyperbola. Therefore, one of the most applied problem for the B-scan GPR image is to transform (or to migrate) the unfocused space-time GPR image to a focused one showing the object's true location and size with corresponding EM reflectivity. The common name for this task is called *migration* or *focusing* [5–18]. In fact, migration methods were primarily developed for processing seismic images [19] and they were also applied to the GPR thanks to the likenesses between the acoustic and the electromagnetic wave equations [7–11]. Kirchhoff's wave-equation [5] and frequency-wave number (ω-k) based [6, 7] migration algorithms are widely recognized and employed. The wave number domain focusing techniques, for instance, were first formulated for seismic imaging applications [7] and then adapted to the contemporary synthetic aperture radar (SAR) imaging practices [12–15]. These algorithms are also named as seismic migration [12, 13] and frequency-wave number (or ω-k) migration [16–18] by different researchers.

In this work, we present a brief review of the B-scan GPR migration/focusing methods that are commonly used by GPR research community. Although we have done some preliminary studies on the performance comparison of only two GPR focusing algorithms earlier [20] by utilizing very limited performance parameters, in this paper, we extend our studies to compare a total of five popular algorithms by assessing different aspects of these algorithms. In Section 2, we address the problem of migration together with the well-known *exploding source concept*. In fact, most migration algorithms make use of this concept as we shall explain in this work. In Section 3, most popular and functional migration methods for the B-scan GPR imaging applications are reviewed. For this purpose, (i) hyperbolic summation, (ii) Kirchhoff's migration, (iii) phase-shift migration, (iv) frequency-wavenumber (or the ω-k) migration, and (v) back-projection method are presented together with the detailed algorithm steps. In Section 4, the success and the performance of these algorithms were compared to each other by testing them over the simulated and the measured B-scan GPR data. Our main objective in this work is to present a qualitative comparison of these popular and widely used algorithms such that the reader can benefit from the performance results over different parameters ranging from resolution to computation time. Section 5 is dedicated to discussions and the conclusion.

2. The Problem of Migration/Focusing

The common goal in a typical GPR image is to display the information of the spatial location and the reflectivity of an underground object. As illustrated in Figure 1, a single point scatterer appears as a hyperbola in the space-time GPR image for the monostatic operation. Since this is the expected image pattern for the B-scan operation, such information can be thought as sufficient if the main goal of the GPR application is just to sense a pipe or comparable objects. However, the information about the scatterer including its depth, its size, and its EM reflectivity can be very important in most GPR applications. Therefore, the hyperbola or dispersion in the space-time B-scan GPR image ought to be converted to a focused one that demonstrates the object's factual location and size together with its scattering amplitude. A focused or migrated image is gathered after the elimination of this hyperbolic type of diffraction or any other kind of dispersion [5–18].

2.1. Exploding Source Model. Most of the migration methods are based on the concept called exploding source model (ESM). In 1985, Claerbout [21] came with the clever idea of thinking of the scattered field at the radar receiver as if it is originated from the source at the target location. Instead of assuming a two-way trip convention of the EM wave, therefore, it is imagined that a fictitious source "explode" at a reference time of $t = 0$ around the target location and send EM wave to the receiver as illustrated in Figure 2. The real data collection scheme is shown in Figure 2(a) where, in fact, the two-way propagation between the radar and the object exists. When the ESM is utilized, however, the collected data is assumed to be originally radiated from the source on the object. Therefore, one-way propagation is assumed in the ESM depicted in Figure 2(b). Since the trip time of the EM wave would be half of the original problem the compensation should be made for the velocity of the EM wave by just dividing it by two to have $v_m = v/2$ where v is the speed of the EM wave within the medium of propagation.

The migration using the ESM is essentially carried out by applying these two actions:

(i) the received signal is extrapolated back to exploding source points;

(ii) the migrated image is realized by forming the back extrapolated EM wave at the time of $t = 0$.

3. Migration/Focusing Methods

In this section, we will briefly review the basic steps of the mostly applied GPR algorithms, namely, the hyperbolic summation, the Kirchhoff migration, the phase-shift migration, the ω-k (Stolt) migration, and the back-projection focusing. Other migration methods will also be mentioned at the end of this section.

3.1. Hyperbolic (Diffraction) Summation. In a typical GPR application, the radar antenna collects the scattered or back-scattered EM wave from the air-to-ground interface and subsurface objects together with many cluttering effects mainly attributed to inhomogeneities within the ground. For the idealistic case, the phase of the scattered signal is directly proportional to the trip time (or distance) that the EM wave possesses if the propagation medium is homogenous. The monostatic backscattered signal from a single point-like scatterer experiences different round-trip distances while the antenna is moving over the surface for the B-scan operation.

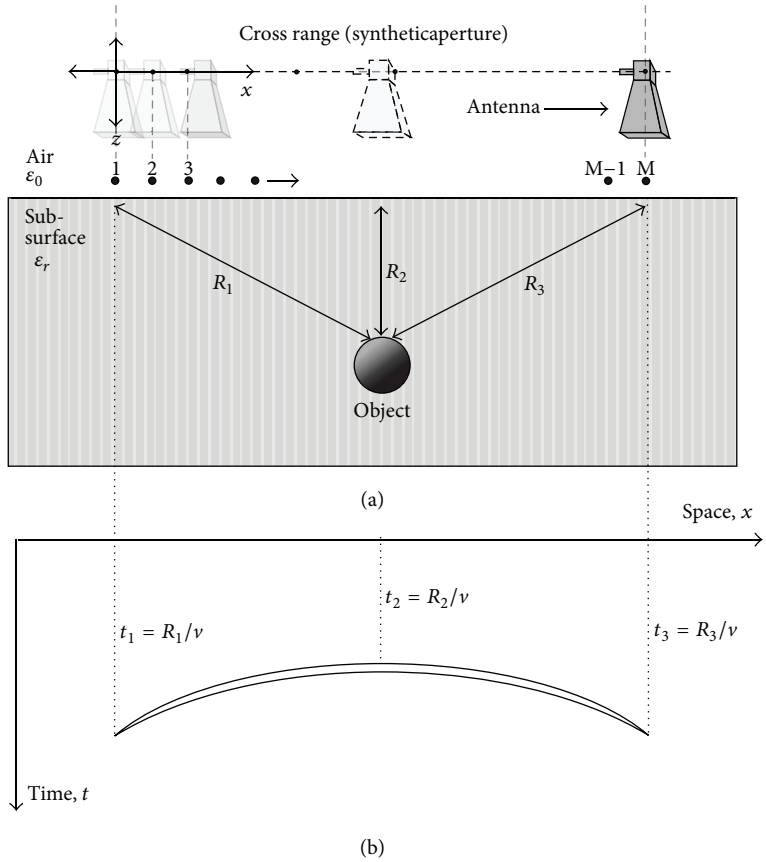


Figure 1: (a) Typical GPR measurement setup (B-scan) and (b) resultant space-time image that contains the well-known hyperbolic aliasing.

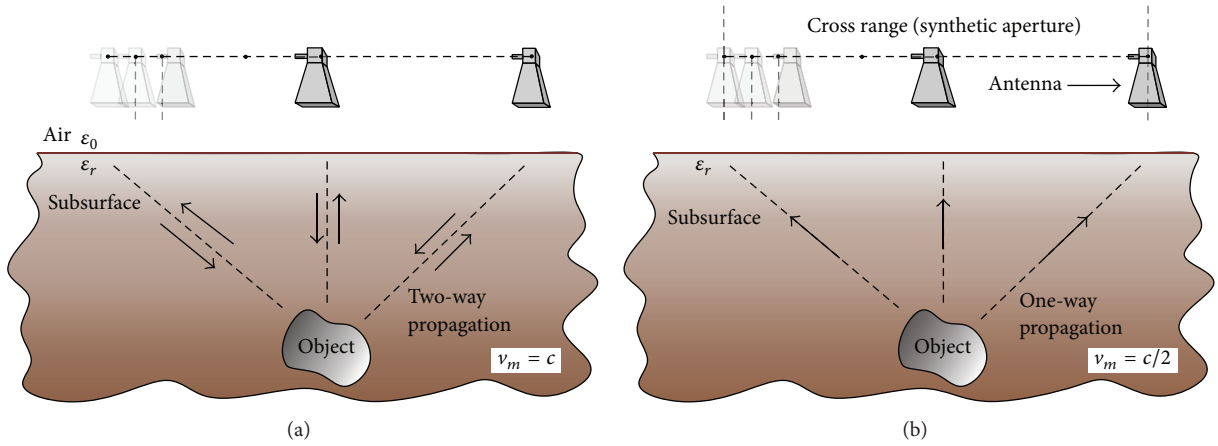


Figure 2: Geometry for (a) B-scan GPR data collection scheme and (b) utilizing "exploding source model."

For each static measurement along the B-scan axis, one-dimensional (1D) range profile (or depth profile) of the subsurface scene is obtained by taking the inverse Fourier transform (IFT) of the frequency-diverse back-scattered signal. After putting all the depth profiles aside, the 2D space-time (or space-depth) B-scan GPR image is obtained. As the GPR antenna has a finite beamwidth, any subsurface object is illuminated for a finite length along the B-scan axis as demonstrated in Figure 1. Therefore, this object shows up as a parabolic hyperbola in the space-time GPR image due to different trip distances of the EM wave between the radar and the illuminated scattering object. The true location of the object is, in fact, at the apex of this hyperbola.

Let us assume a perfect point scatterer situated at (x_o, z_o) in the 2D plane where x-axis corresponds to the scanning direction and the z-axis is the depth as illustrated in Figure 1. If the propagating medium is homogeneous, the parabolic hyperbola in the GPR image can be characterized by the

following equation when the radar is moving on a straight path along X-axis. Consider

$$\mathbf{R} = \sqrt{z_o^2 + (\mathbf{X} - x_o)^2}. \quad (1)$$

In the above equation, $\mathbf{X}$ is the synthetic aperture vector along the B-scan and $\mathbf{R}$ represents the path length vector from the antenna to the scatterer. Assuming that the resultant B-scan GPR image can be regarded as the contribution of finite number of hyperbolas that correspond to different points on the object(s) below the surface, the following methodology can be applied to migrate the defocused image structures, that is, hyperbolas to the focused versions [22].

(i) For each pixel point, (x_i, z_i) on the 2D original B-scan space-depth GPR image matrix, find the associated hyperbolic template using (1) and trace all the pixels under this template.

(ii) Record the image data for the traced pixels under that template. At this stage of the algorithm, we have 1D field data E^s, whose length N is the same as the number of sampling points along the synthetic aperture X.

(iii) Then, calculate the root-mean-square (rms) of the entire energy contained inside this 1D complex field data as follows:

$$\begin{aligned}\{\text{rms at } (x_i, z_i)\} &= \sqrt{\frac{\left(|\mathbf{E^s}|^2 \left|(\mathbf{E^s})^*\right|^2\right)}{N}} \\ &= \frac{1}{\sqrt{N}} \sum_{i=1}^{N} |\mathbf{E_i^s}|^2.\end{aligned} \quad (2)$$

(iv) The calculated rms figure is recorded on the new image matrix at the point (x_i, z_i). This process is reiterated until all pixels on the original GPR image are passed through the algorithm.

3.2. Kirchhoff's Migration. *Kirchhoff's migration* (KM), also known as *reverse-time wave equation migration*, is mathematically equivalent to hyperbolic summation method with some correcting factors included in the solution [22]. In Kirchhoff's migration procedure, the aim is to find the solution to the following scalar wave equation for the wave function $\varphi(x, z, t)$ within the propagating medium:

$$\left(\frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial z^2} - \frac{1}{v_m^2}\frac{\partial^2}{\partial t^2}\right)\varphi(x, z, t) = 0. \quad (3)$$

Here, v_m is the velocity of the EM wave within the propagating medium and taken as $v/2$ by utilizing the "exploding source" concept. The solution to this differential equation is available for the far-field approximation from the *Kirchhoff integral theorem* [23] as

$$P(x, z, t) = \frac{1}{2\pi} \int_{-\infty}^{\infty} \left(\frac{\cos\theta}{v_m R} \cdot \frac{\partial}{\partial t} P\left(x, z, t - \frac{r}{v_m}\right)\right) dx. \quad (4)$$

Here, θ is the angle of the incident wave to the depth axis (z) and $r = ((x - x_m)^2 + z^2)^{1/2}$ is the path from the target point at (x, z) to the observation point at $(x_m, 0)$.

In KM method, the following correction factors that are not considered in the hyperbolic summation are accounted.

(i) Compensation for the spherical spreading is taken into account. For this purpose, a correction factor of $1/\sqrt{v_m r}$ is used for the 2D propagation of the EM wave, whereas this factor becomes $1/(v_m r)$ for the 3D propagation.

(ii) The directivity factor $\cos\theta$ is also considered. This factor corrects the diffraction amplitudes.

(iii) The phases and amplitudes of the wave are also corrected. The phase is corrected by $\pi/2$ and $\pi/4$ for 2D and 3D propagation cases, respectively. The amplitude of the EM wave is corrected with a factor proportional to the square of the frequency for the 2D propagation case and to the frequency for the 3D propagation case [23–25].

3.3. Phase-Shift Migration. The phase-shift migration (PSM) method is first introduced and applied by Gazdag [6]. Similar to Kirchhoff's migration, this method also utilizes the ESM concept [26]. In brief, the algorithm iteratively puts a phase-shift to migrate the wave field to the exploding time of $t = 0$ such that all the scattered waves are drawn back to object site to have a focused image.

The main aim of the PSM algorithm is to calculate the wave field at $t = 0$ by extrapolating the downward (z-directed) EM wave with the phase factor of $\exp(jk_z z)$.

The PSM algorithm can be briefly summarized via the following steps.

(i) First, 2D measured raw dataset in frequency wave-number or (ω, k_x) domain is multiplied by a phase-shift factor, K, along the depth axis (or z-axis) as given below

$$K = e^{jk_z \cdot \Delta z}. \quad (5)$$

This factor can also be written in terms of wave-number along the data collecting axis (k_x) as

$$\begin{aligned}K &= \exp(jk_z \cdot \Delta z) \\ &= \exp\left(j\sqrt{k^2 - k_x^2} \cdot \Delta z\right) \\ &= \exp\left(jk\sqrt{1 - \left(\frac{k_x}{k}\right)^2} \cdot \Delta z\right).\end{aligned} \quad (6)$$

Putting $k = \omega/v_m$ (where ω is the angular frequency and v_m is the speed of wave inside the propagation medium) into (5), one can get

$$K = \exp\left(j\frac{w}{v} \cdot \Delta z\sqrt{1 - \left(\frac{v_m \cdot k_x}{\omega}\right)^2}\right). \quad (7)$$

In the above equation, the incremental depth parameter Δz is proportional to the time sampling interval Δt of the input

data via $\Delta z = v_m \cdot \Delta t$. After this modification, the final form of the phase-shift factor becomes

$$K = \exp\left(j\omega \cdot \Delta t \sqrt{1 - \left(\frac{v_m \cdot k_x}{\omega} \right)^2} \right). \tag{8}$$

2D measured raw dataset $E^s(\omega, k_x)$ is multiplied by K along the depth axis for the time steps of Δt.

(ii) By utilizing the ESM concept, the imaging task is accomplished by taking the inverse Fourier transform (IFT) of the $E^s(\omega, k_x)$ after selecting the time variable as $\Delta t = 0$. Therefore, only single FT operation is required at one point (when $\Delta t = 0$) for the focused image.

After updating the factor K for every value of Δt and ω, we have the data in 3D (ω, k_x, k_z) domain as

$$E^{s\prime}(\omega, k_x, k_z) = K \cdot E_s(\omega, k_x). \tag{9}$$

The new dataset $E^{s\prime}(\omega, k_x, k_z)$ is summed up along the frequency axis and indexed for different values of Δt as

$$E^s(k_x, k_z, t) = \sum_w E^{s\prime}(\omega, k_x, k_z). \tag{10}$$

(iii) Setting $\Delta t = 0$ and taking the 2D IFT with respect to k_x and k_z, the focused the image in the (x, z) domain can be obtained as given below:

$$E^s(x, z) = \text{IFT}\left\{ E^s(k_x, k_z) \right\}. \tag{11}$$

3.4. Frequency-Wavenumber (Stolt) Migration. Frequency-wavenumber (ω-k) migration, also known as *Stolt migration* or the f-k migration, utilizes the ESM idea and the scalar wave equation [7]. The algorithm behind the frequency-wavenumber method works faster than the previously presented migration methods. The ω-k migration method has proven to be working well for the constant-velocity propagation mediums [6, 7]. The solution of the ω-k migration can be rewritten to be the same as the solution of the Kirchhoff migration [18]. Below is the brief explanation of the algorithm.

The algorithm begins with the 3D scalar wave equation for the wave function $\varphi(x, y, z, t)$ within the constant-velocity propagation medium

$$\left(\frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2} + \frac{\partial^2}{\partial z^2} - \frac{1}{v_m^2} \frac{\partial^2}{\partial t^2} \right) \varphi(x, y, z, t) = 0. \tag{12}$$

In the Fourier space, spatial wave-numbers and the frequency of operation are related with the following equation:

$$k_x^2 + k_y^2 + k_z^2 = k^2 = \frac{\omega^2}{v_m^2}. \tag{13}$$

Stratton [27] demonstrated that any given wave function can be written as the summation of infinite number of plane wave functions, say $E(k_x, k_y, \omega)$, as

$$\varphi(x, y, z, t) = \left(\frac{1}{2\pi} \right)^{3/2} \iiint_{-\infty}^{\infty} E(k_x, k_y, \omega) \times e^{-j(k_x x + k_y y + k_z z - \omega t)} dk_x dk_y d\omega. \tag{14}$$

For the GPR operation, the scattered field is taken to be measured on the $z = 0$ plane above the surface. When carefully treated, the above equation offers a Fourier transform pair where $e(x, y, t)$ can be regarded as the time-domain measured field on the $z = 0$ plane. Consider

$$\varphi(x, y, 0, t) \triangleq e(x, y, t) = \left(\frac{1}{2\pi} \right)^{3/2} \iiint_{-\infty}^{\infty} E(k_x, k_y, \omega) \times e^{-j(k_x x + k_y y - \omega t)} dk_x dk_y d\omega. \tag{15}$$

It is very important to notice that this equation designates a 3D forward FT relationship between $e(x, y, t)$ and $E(k_x, k_y, \omega)$ for the negative values of time variable, t. Then, the inverse FT can be dually defined in the following way:

$$E(k_x, k_y, \omega) = \left(\frac{1}{2\pi} \right)^{3/2} \iiint_{-\infty}^{\infty} e(x, y, t) \times e^{j(k_x x + k_y y - \omega t)} dx\, dy\, dt. \tag{16}$$

Afterwards, we now use the ESM to focus the image by setting $t = 0$ in (15) and using

$$E(k_x, k_y, \omega) = e^{jk_z z} E(k_x, k_y, \omega, z = 0). \tag{17}$$

Therefore, one can get the time-domain measured field via

$$e(x, y, z, 0) = \left(\frac{1}{2\pi} \right)^{3/2} \iiint_{-\infty}^{\infty} E(k_x, k_y, \omega) \times e^{-j(k_x x + k_y y + k_z z)} dk_x dk_y d\omega. \tag{18}$$

The above equation presents a focused image. However, the data in (k_x, k_y, ω) domain should be transformed to (k_x, k_y, k_z) domain to be able to use the FFT. Therefore, a mapping procedure from ω domain to k_z domain is required for fast processing. The relationship between the ω- and -k_z and $d\omega$- and -dk_z can be easily obtained from (13) as

$$\omega = v_m \left(k_x^2 + k_y^2 + k_z^2 \right)^{1/2} \tag{19a}$$

$$d\omega = \frac{v_m^2 k_z}{\omega} dk_z. \tag{19b}$$

Substituting these equations into (18), one can obtain the following:

$$e(x, y, z) = \left(\frac{1}{2\pi} \right)^{3/2} \iiint_{-\infty}^{\infty} \frac{v_m^2 k_z}{\omega} E^m(k_x, k_y, k_z) \times e^{-j(k_x x + k_y y + k_z z)} dk_x dk_y dk_z. \tag{20}$$

Here, $E^m(k_x, k_y, k_z)$ is the mapped version of the original data $E(k_x, k_y, \omega)$. After this mapping, the new data set does

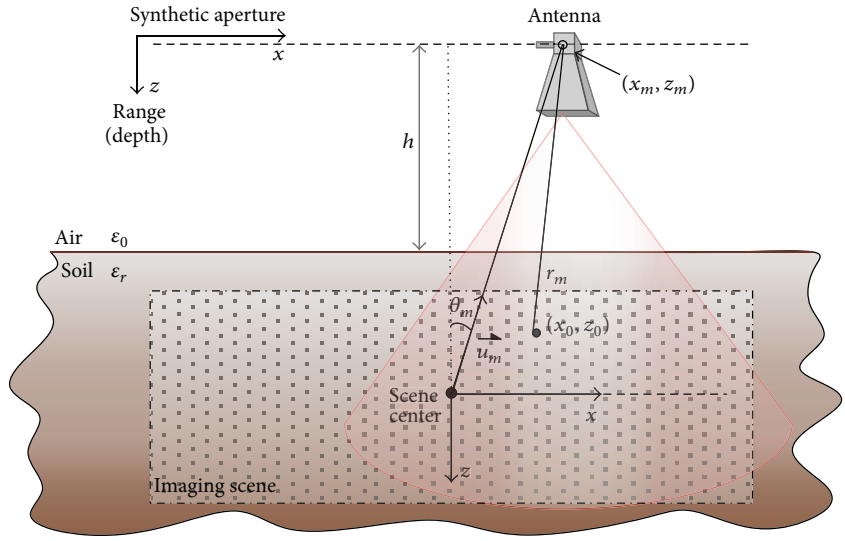


Figure 3: The B-scan geometry of the 2D monostatic GPR application.

not lie on the uniform grid due to nonlinear feature of the transformation. Therefore, an interpolation procedure should also be applied to be able to use the FFT for fast processing of the collected dataset. Equation (20) suggests a well-focused image of the subsurface region that may contain a finite numbers of scatterers. Since the FFT routine is utilized, the GPR image in the ω-k migration is obtained quite fast. It is also important to express that the mapped dataset is scaled by the factor of $v_m^2 k_z/\omega$. This scaling is sometimes called the "*Jacobian transformation from* ω *to* k_z."

The above focusing equation is valid for the 3D GPR geometry or the C-scan problem. In most subsurface imaging problems, however, the raw data is collected in 2D space, that is, space-time (space-depth) or space-frequency. Therefore, the focusing equation in (20) can be easily reduced to 2D B-scan GPR problem in the space-depth domain via the following equation:

$$e(x,z) = \left(\frac{1}{2\pi}\right) \iint_{-\infty}^{\infty} \frac{v_m^2 k_z}{\omega} E^m(k_x, k_z) \times e^{-j(k_x x + k_z z)} dk_x dk_z. \tag{21}$$

3.5. Back-Projection Based Migration. The last category of the GPR focusing algorithms is based on the tomographic principles used in medical imaging and collectively termed as the back-projection (BP) algorithms. Since its first formulation for the 2D monostatic SAR processing [28], the BP algorithm has gathered an increasing interest in radar community thanks to its distinct features that are proved to be very useful for various SAR imaging applications. For example, the algorithm does not require a straight and uniformly sampled synthetic scan aperture due to its serial processing nature. More clearly, each 1D range profiles are serially processed and spread or back-projected over the entire 2D image independently. This sequential processing means "real-time" operation capability and hence the imaging process can begin before acquiring the entire synthetic aperture data. Furthermore, the specific subsections of the region to be imaged can be easily selected to investigate these subsections more closely. In applications where the approximate location of the target is known a priori, the detailed image of the region around this target can be easily formed by the algorithm.

Before briefly explaining the algorithm, let us first consider the B-scan, monostatic GPR geometry shown in Figure 3. A radar antenna, situated at a height of h from the ground, transmits a frequency-diverse waveform at each distinct point m along the synthetic axis. The relative permittivity of the subsurface medium is denoted by ε_r and the reflectivity function of the scatterers is represented by $e(x,z)$. The instantaneous position of the antenna (x_m, z_m) is defined by a unit vector $\mathbf{u_m}$ pointing from the scene center towards this location. The corresponding view-angle θ_m is defined as the angle between the unit vector $\mathbf{u_m}$ and the depth axis z. Assuming a stepped-frequency continuous waveform (SFCW) transmission, the back-scattered signal at a specific observation angle can be written as

$$S_{\theta_m}[k_r] = \iint_{-\infty}^{\infty} e(x,z) \exp(-jk_r r_m)\, dx\, dz, \tag{22}$$

where k_r is the wavenumber defined for the two-way propagation as $k_r = 4\pi f/v$, f is the frequency, v is the speed of the propagation, and r_m is the range from the instantaneous antenna location (x_m, z_m) to the points (x_0, z_0) within the imaging scene. The range profile of the illuminated scene can be obtained by employing 1D IFT to (22) and can be mathematically conveyed as

$$s_{\theta_m}(r) \equiv \mathrm{IFT}\left\{S_{\theta_m}(k_r)\right\} = \iint_{-\infty}^{\infty} e(x,z)\, \delta(r_m - r)\, dx\, dz \tag{23}$$

which is nothing but the Radon transform of the scene.

The derivation of the BPA [29] begins with the implementation of IFT of the scattering function $e(x, z)$ written in Cartesian coordinates as

$$e(x,z) = \iint_{-\infty}^{\infty} E(k_x, k_z) \exp\left[j(k_x x + k_z z)\right] dk_x dk_z, \quad (24)$$

where $E(k_x, k_z)$ is the 2D FT of $e(x, z)$. Equation (24) can be reformed to be rewritten in the polar coordinates (k_r, θ_m) as follows:

$$e(x,z) = \int_{-\pi}^{\pi} \int_{0}^{\infty} E(k_r, \theta_m) \exp(jk_r r_m) k_r dk_r d\theta_m. \quad (25)$$

Now, the projection-slice theorem [30] can be used to relate the target's FT $E(k_x, k_z)$ to the collected measured data $S_\theta(k_r)$. For the 2D problem, the theorem essentially states that 1D FT of the projection at the angle θ represents the slice of the 2D FT of the projected (original) scene at the same angle; that is, $S_\theta(k_r) = E(k_r, \theta)$. Therefore, the sampled representation of $E(k_x, k_z)$ can be obtained from the FT of the projections $S_\theta(k_r)$ measured at several observation aspects. By the help of this principle, (25) becomes

$$e(x,z) = \int_{-\pi}^{\pi} \left[\int_{0}^{\infty} S_{\theta_m}(k_r) \exp(jk_r r_m) k_r dk_r \right] d\theta_m. \quad (26)$$

The bracketed integral term in (26) can be regarded as the 1D IFT of a function $Q_{\theta_m}(k_r) = S_{\theta_m}(k_r)k_r$ calculated at r_m. Defining $q_{\theta_m}(r)$ as the IFT of this function, (26) can be represented as

$$\rho(x,z) = \int_{-\pi}^{\pi} q_{\theta_m}(r_m) d\theta_m. \quad (27)$$

Equation (27) is the final focused image of the 2D filtered back-projection algorithm. For the SFCW system, the execution of the algorithm can be summarized as follows.

(1) Preallocate an image matrix of zeros $e(x, z)$ to hold the values of the scene reflectivity.

(2) Multiply the acquired spatial-frequency data $S_{\theta_m}(k_r)$ with k_r.

(3) Take 1D IFT of the result to obtain $q_{\theta_m}(r)$ which represents the filtered version of the range profile $s_{\theta_m}(r)$.

(4) For each pixel position in the image data, evaluate the corresponding range value r_m and acquire its $q_{\theta_m}(r_m)$ value by the help of an appropriate interpolation method.

(5) Iteratively add interpolated data values to $e(x, z)$.

(6) Repeat the above steps from 2 to 5 to cover all the observation angles θ_m.

3.6. Some Other Methods. In addition to the above mentioned common migration methods, there are many other techniques that have been introduced and studied by different GPR researchers. Fisher et al. [31] applied reverse-time migration procedure for GPR profiles. Capineri et al. [32] applied a technique based on Hough transformation to the B-scan GPR data to attain better resolved images of pipe structures. Leuschen and Plumb [9] and Morrow and van Genderen [33] realized back-propagation procedures that rely on finite difference time-domain (FDTD) reverse-time focusing methods to resemble the focusing matter in GPR images.

4. Performance Comparison of the Algorithms

Performance of the algorithms is evaluated by the help of following various parameters that are resolution, integrated sidelobe ratio, signal to clutter ratio, and computation speed.

Resolution. GPR measurement can provide fine resolutions both in depth and azimuth resolutions. The term depth (range) corresponds to the line of sight distance from the radar to the target to be imaged. The term azimuth (transverse range, cross range) is used for the dimension that is perpendicular to range or parallel to the radar's along-track axis [34]. The depth resolution is defined as an ability of the radar equipment to separate two reflecting objects on a same line of sight, but at different ranges from the antenna. And the azimuth resolution can be defined as a capability of the separation of two reflecting objects on the same ranges, but different aspects from the radar. In GPR operation, the high resolution in depth is obtained by utilizing a transmitted signal of wideband. Fine azimuth resolution is achieved by coherently processing the target's electromagnetic scattering measured at different aspects while the radar is moving along a straight path. However, the measured resolutions after the postprocessing are dependent on the focusing ability of the migration algorithm. That is why the resolution performance of the algorithms is tested by measuring the −4 dB contours of a point scatterer (whose coordinate is $x = 0$ m and $z = 1.5$ m) in the resultant images of simulations. Normally, algorithms with better focusing abilities are expected to produce lower resolved images.

Integrated Sidelobe Ratio (ISLR). Since ISLR is defined as the ratio of the total energy within the sidelobes to the peak energy of the main lobe [35], it can be regarded as an important parameter when assessing the focusing abilities of migration algorithms [36]. Based on Sanchez's definition of ILSR, it is the ratio of the −3 dB width of the main lobe to the rest of the energy within all lobes. Therefore, the ISLR of a 2D image function $I(x, y)$ can be formulated as follows:

$$\mathrm{ISLR} = 10\log_{10}\left(\frac{\int_{-3\,\mathrm{dB}} I dx' dy'}{\int_{-\infty}^{+\infty} I dx' dy' - \int_{-3\,\mathrm{dB}} I dx' dy'} \right), \quad (28)$$

where $\int_{-3\,\mathrm{dB}} I dx' dy'$ is the energy within the −3 dB width of the main lobe and $\int_{-\infty}^{+\infty} I dx' dy'$ is the total energy.

Table 1: Performances of focusing algorithms over different parameters for the simulated experiments.

Algorithm	Computation time (s)	Depth resolution (cm)	Azimuth resolution (cm)	ISLR (dB)
HSA	46.32	6.78	12	−13.15
KMA	9,372.70	5.43	4	−12.17
PSA	2.13	8.70	10	−13.02
ω-kA	0.41	6.30	4	−12.75
BPA	4.29	7.70	5.8	−6.13

Signal to Clutter Ratio (SCR). Signal to clutter ratio (SCR) is also a crucial parameter when evaluating the quality of the focusing ability. In a well-focused image, the clutter/noise level should be much lower than the signal level to have a well-contrasted image. The SCR can be defined as the ratio of the received target signal power to the received clutter power as shown below:

$$\mathrm{SCR} = 10\log_{10}\left(\frac{P_{\mathrm{tar}}}{P_{\mathrm{tot}} - P_{\mathrm{tar}}}\right), \tag{29}$$

where P_{tar} is the received target signal power and P_{tot} is the total power within the image.

Computation Speed. The processing time of the computation of the algorithms can be crucial if the GPR application requires processing of vast data such as scanning and cleaning a minefield.

4.1. Comparison over Simulation. The migration algorithms were first tested through an imaging simulation performed in MATLAB. Considering the classical B-scan geometry shown in Figure 4, the subsurface was assumed to have a completely homogeneous soil medium structure (i.e., constant-velocity medium) with a dielectric constant of 2.4. The medium is also assumed to be nonmagnetic; that is, $\mu_r = 1$. The monostatic operation was considered and the −3 dB beamwidth of the antenna was assumed to be large enough such that all targets to be imaged fall inside the beam for the whole aperture with constant antenna gain. The electric field data of the isotropic point scatterers with locations depicted in Figure 4 were then collected along a straight path ranging from $x = -2.5$ m to $x = 2.5$ m and with 2 cm steps. At each spatial point, the frequency response of the subsurface was acquired for the 1.25 GHz–3.75 GHz frequency range that was uniformly sampled at 168 points. Therefore, 251 × 168 spatial-frequency B-scan data $E(x, f)$ were generated for the investigated scene. The frequency data were subsequently preprocessed by applying a Hanning window for sidelobe control and 4 times zero padding for interpolation purposes.

Figure 5(a) shows the raw, unfocused B-scan image simply formed by applying a 1D inverse Fourier transform to the collected data, $E(x, f)$. As expected, the target reflections are observed as hyperbolas with different intensities and curvature slopes. To collapse these hyperbolic signatures at the apex of the hyperbolas, the migration algorithms were applied and the corresponding obtained results were given in Figures 5(b)–5(f). As a quick interpretation, it can be noticed from all images that the scattering mechanisms are almost concentrated around the exact locations of the targets.

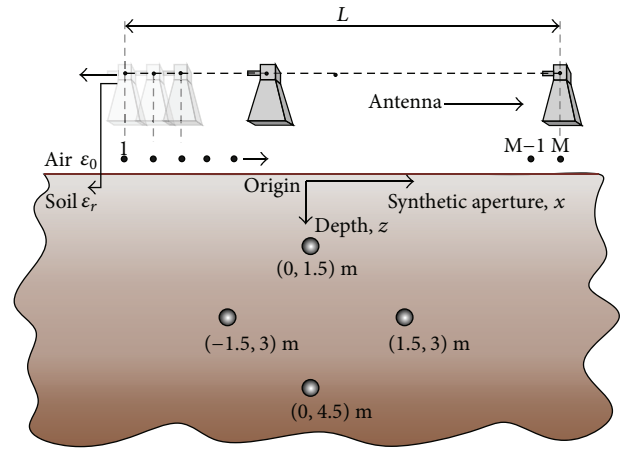


Figure 4: Simulation geometry of the B-scan imaging example.

To compare the performances of the algorithms, Table 1 lists the success merits of the algorithms including the depth resolution, the azimuth resolution, the ISLR, and the computational time. By looking at the resolution outcomes in both depth and azimuth direction, KM and ω-K algorithms seem to provide better resolved images. On the other hand, KMA is the worst in terms of computation time while ω-kA is the fastest among all others. In terms of the ISLR, BPA is superior to any other algorithm by providing a sharp ISLR that is at least 6 dB better than the others. When comparing all parameters, one can conclude that the BPA and ω-kA seem to have a little bit better in terms of the migration ability.

When the resultant focused images of these algorithms are visually compared in Figure 5, the KM, the BPA, and the ω-kA look more successful than others as the results of the HSA and PSA expose some image degradations. If those well-localized images of Figures 5(c), 5(e), and 5(f) are compared, some minor differences can even be discerned between each other. For example, the target reflections in the BP migrated image have more regular pattern and also have stronger intensities. Also, the images for the KM and the ω-kA have some relatively higher sidelobe levels around the top and bottom targets. Additionally, if the image for the ω-kA is carefully checked (see Figure 5(e)), the depth of the bottom target is also shown to be mapped to $z = 4$ m which indicates a little deviation from its true value of $z = 4.5$ m.

By evaluating the GPR images in Figure 5, we can compare the focusing algorithms against each other as follows. While the ω-kA may experience some problems in imaging deep targets with low reflectivities, successful imaging of

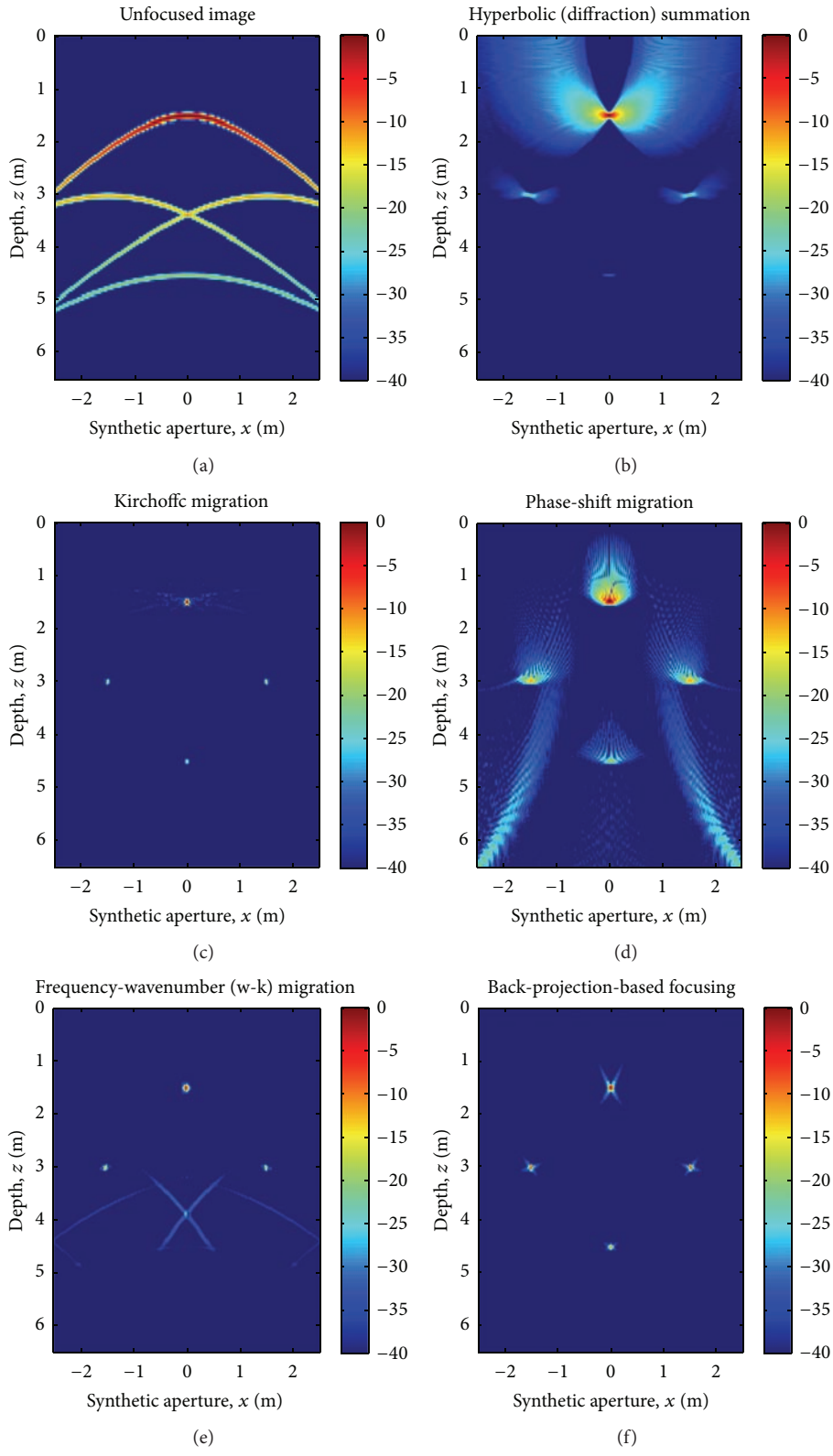


Figure 5: Simulation results for the −40 dB dynamic range.

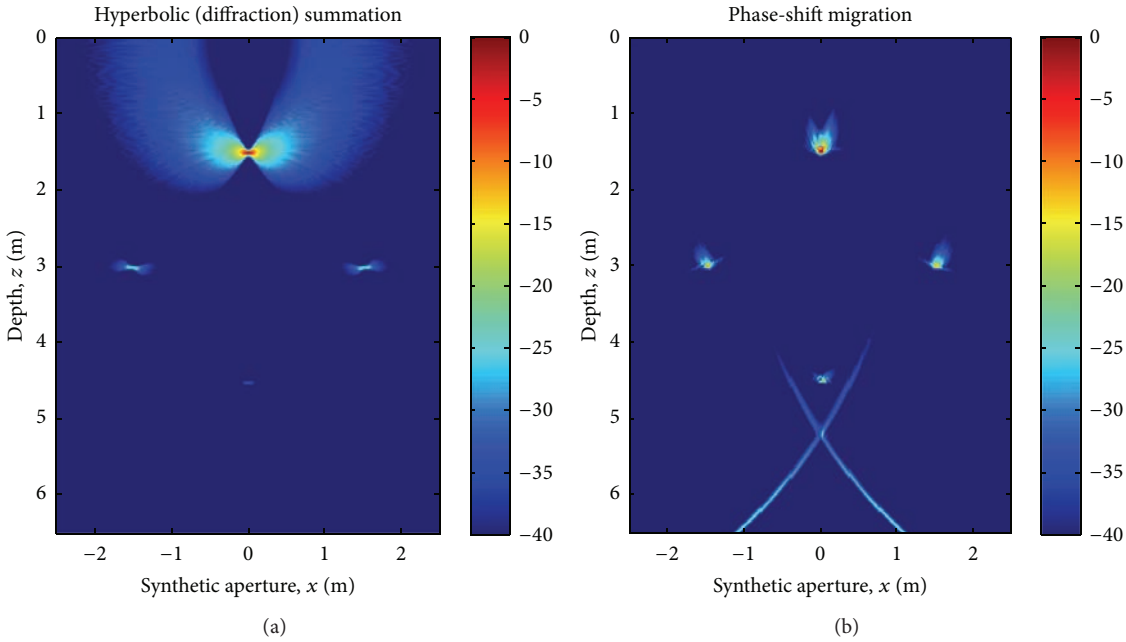


Figure 6: Results of the hyperbolic summation and Kirchhoff's migration algorithms for a decreased frequency sampling interval (i.e., frequency bandwidth $B = 4.5$ GHz with 302 sampling points).

shallow targets can be problematic for the KM. Secondly, the images for the HSA and the PSA look considerably poorer than those of the other algorithms. This can be seen from the significantly defocused image that corresponds to PSA (see Figure 5(d)). This image also has some extra undesired scattering features such as blurring and spreading. On the other hand, the image for the HSA exhibits a major degradation only around the uppermost target. The other targets are observed to be almost well focused.

To comprehend the sources of abovementioned defocusing effects for these two algorithms, another simulation was performed by utilizing different simulation parameters. For this simulation, the total number of frequency samples was increased to 302 points. The bandwidth was then correspondingly changed to $B = 4.5$ GHz to satisfy the same unambiguous range of the previous simulation, that is, 6.5 m. Corresponding results for the simulation are demonstrated in Figure 6. It is shown from the Figure 6(a) that the HSA is not affected from the variation of the sampling points along the depth direction. The migrated image is similar to the previous result given in Figure 5(b), except for the definite resolution enhancement thanks to wider bandwidth. In any case, HSA is shown to have difficulty only in providing a reasonable representation of the uppermost target along the spatial direction. Although this aliasing is also somewhat observed for the deeper targets, they can also be regarded as well focused within the dynamic range of −40 dB. As stated before, this spatial aliasing for the shallower target is also slightly observed in the result of the KA as given in Figure 5(c). Thus, noting the similarities between these algorithms, this fact can be assumed to stem from the characteristics of these methods. Nevertheless, this spatial aliasing can be avoided by utilizing different implementations, such as the ones that incorporate more points along a hyperbolic trajectory, rather than using a single point during integration.

The PSA result seen in Figure 6(b) conversely shows drastic improvement when compared to the previous one shown in Figure 5(d). All of the four scatterers are shown to be well localized for the increased value of the frequency sampling points. This notable change can be attributed to the algorithms' iterative processing nature along the depth direction. Since the PSA first records the data at the surface and repeatedly multiply this data with a phase factor for each downward range points, the short sampling intervals would result in better quality images. Hence, this synthetic example illustrates that the PSA is highly sensitive to the employed frequency sampling interval. Lastly, due to the algorithmic similarities between the PSA and the ω-kA, the corresponding images given in Figures 6(b) and 5(e) are shown to have similar distortion patterns especially seen at the bottom of these images.

Finally, the computational efficiency of the algorithms is tested by looking at the simulation runtime and the required memory during implementation. Based on our evaluation, the computational efficiency of the algorithms is listed from the best to the worst as follows: the ω-kA, the PSA, the BPA, the HSA, and the KM.

4.2. Comparison over Measurement. The performance of the focusing algorithms is also tested by the help of measured data sets. For this purpose, two different experiments were conducted to better understand the differences between the focusing capabilities of the algorithms for real datasets. These experiments were conducted by the help of the Agilent E5071B ENA Vector Network Analyzer (VNA) that can

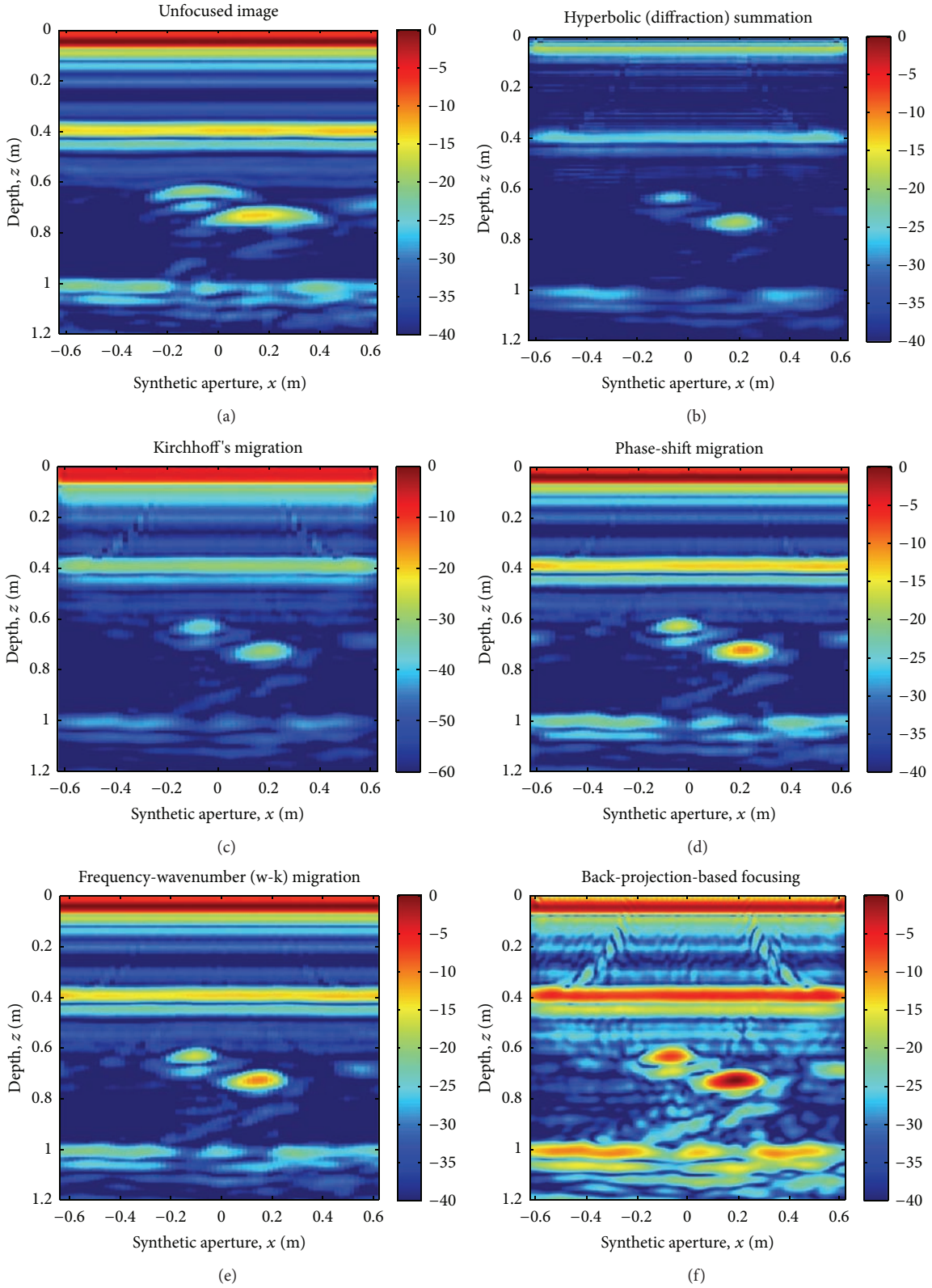


Figure 7: Results for the real soil Experiment 1. Note the different dynamic range (i.e., −60 dB) for the Kirchhoff migration result.

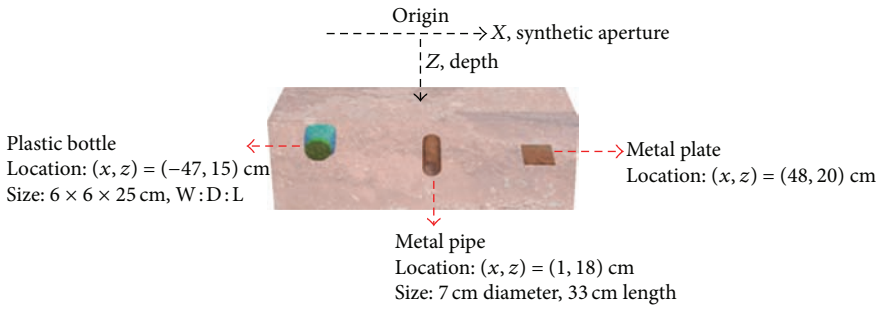


Figure 8: Geometric representation of the targets used in Experiment 2.

generate SFCW signals within the frequency range of 0.8–8.5 GHz. In both experiments, a C-band double-ridged pyramidal rectangular horn antenna was used as a monostatic transceiver.

4.2.1. Experiment Number 1. In the experiment, we have constructed a test bed by building a big wooden pool with a size of 190 cm × 100 cm × 80 cm and filled it with homogeneous and dry sand material. The dielectric constant of the sand was measured to be almost constant around 2.4 for the frequency range between 1 GHz and 8 GHz. It is also assumed that the sand material has the unit magnetic permeability for the frequencies of operation. Two cylindrical metal rods with different sizes were selected as targets in the scene. The thin rod with 4.5 cm in diameter and 45 cm in length was buried at ($x = -12$ cm, $z = 65$ cm) and a thick rod with 6 cm in diameter and 32 cm in length was buried at ($x = 18$ cm, $z = 75$ cm). A B-scan measurement was accomplished by moving the antenna along the perpendicular direction of the target axes and by spanning a synthetic aperture length of $L = 1.34$ m sampled at 63 spatial points. The stepped frequency response of the medium was then obtained for a bandwidth of 0.8–4.5 GHz that was uniformly sampled at 751 points.

After collecting the data, the classical space-depth B-scan GPR image is obtained as shown in Figure 7(a) from which the unfocused target signatures can be clearly seen. The governing scattering mechanism from air-ground boundary is also easily detected around at $z = 40$ cm and seen throughout the entire synthetic aperture. Then, we apply the above presented migration algorithms in aiming at getting a better focused image. The resultant migrated images are displayed in Figures 7(b) to 7(f) wherein the target responses are seen to be more localized around their correct locations. To better compare the algorithms against each other, all resultant focused images were displayed within the same dynamic range of −60 dB. The focusing performance of all algorithms seems to be very similar to each other as the tails (or the sidelobes) of the targets' images are comparable dispersion features. In terms of the image contrast, the ω-kA and the PSA outperforms the BPA, the HSA, and the KM as it can be clearly deducted from the images. For this experiment, the numerical noise generated by the iterative implementation of the BPA seems to be very high when compared to other algorithms. In fact, the real data contains infinite number of scatterers (with high or low reflectivities) under the ground; it seems that the BPA is more sensitive to reflectivity amplitude than the others.

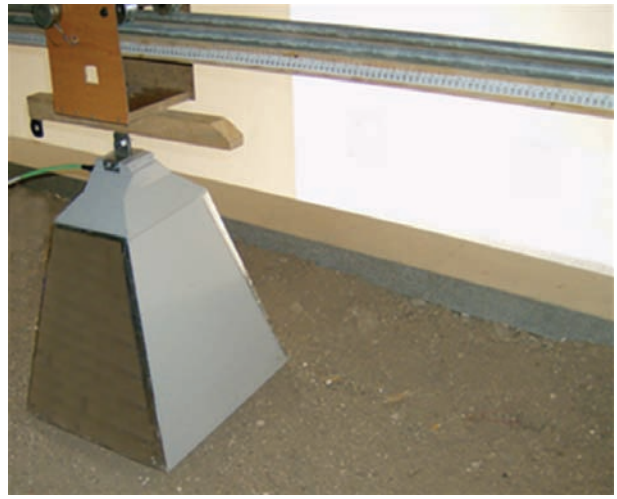


Figure 9: A scene from real soil measurements.

4.2.2. Experiment Number 2. In the second experiment, the algorithms were tested under a more heterogeneous soil in another test bed of the indoor environment. The targets were selected as a water-void target, a metal pipe, and a metal plate whose sizes and burial locations were as depicted in Figure 8. The synthetic aperture length was set to 134 cm for a total of 68 discrete points and the frequency was changed from 0.8 GHz to 5 GHz for 501 discrete points. A picture during the experiment is presented in Figure 9.

After collecting the data, the raw image obtained by taking the IFT of the measured back-scattering data is shown in Figure 10(a). As expected, the image is unfocused around the buried objects. After applying the migration algorithms, the focused images of the three buried objects are acquired as seen in Figures 10(b) to 10(f). For this experiment, very similar features were observed as in the case of the first experiment. Again, focusing performances of the algorithms are alike as the sidelobe contours of target's images are comparable. The contrast performances of the images also yield

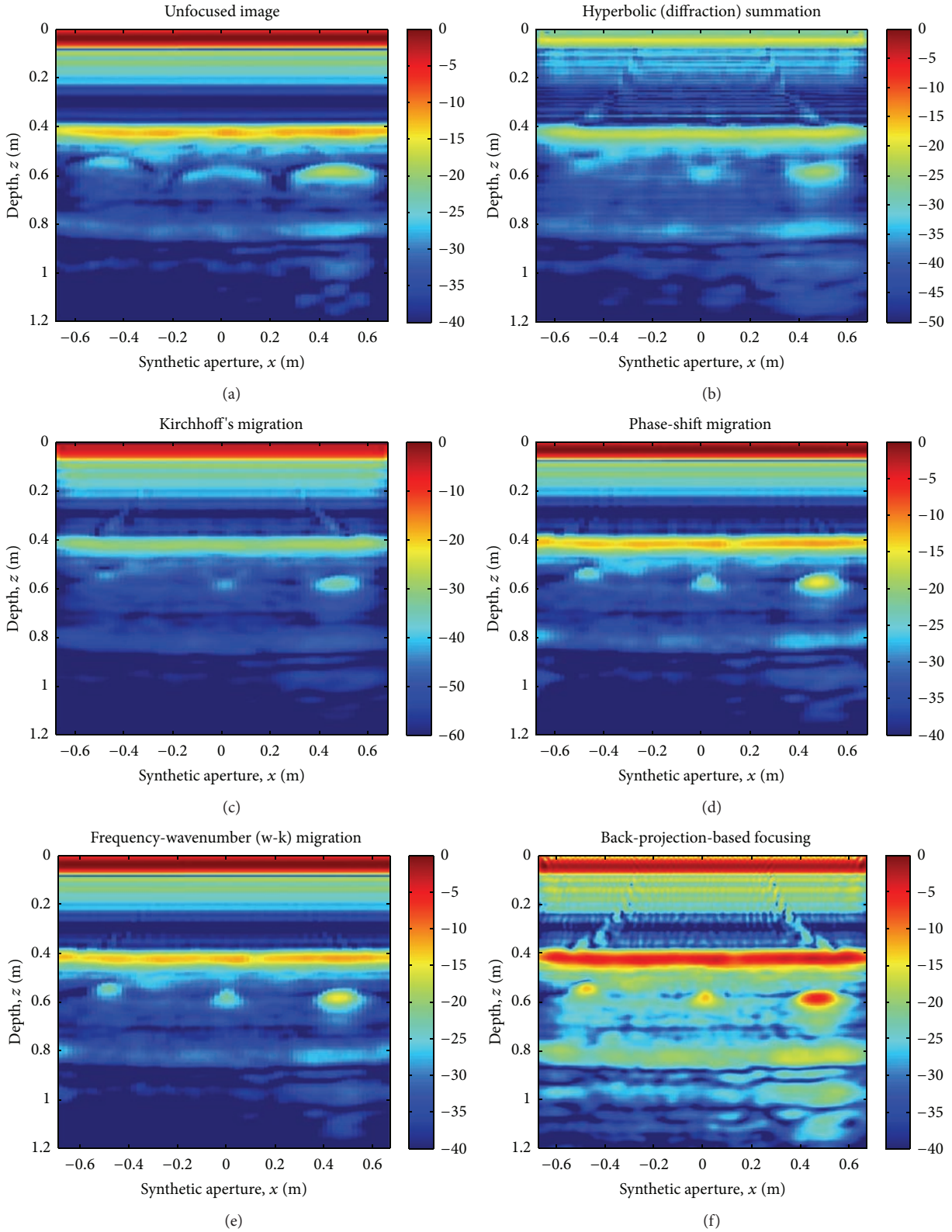


Figure 10: Results for the real soil Experiment 2. Note the different dynamic ranges (i.e., −50 and −60 dB) for the hyperbolic summation and Kirchhoff's migration results.

Table 2: Performances of focusing algorithms over different parameters for the measured experiments.

	Experiment number 1 results		Experiment number 2 results	
	SCR (dB)	Time (s)	SCR (dB)	Time (s)
HSA	−6.27	11.12	−12.48	7.61
KMA	−4.55	5,723.80	−9.47	26,76.30
PSA	−5.41	13.99	−10.40	3.74
ω-kA	−5.66	0.57	−9.92	0.34
BPA	−4.48	1.75	−9.01	2.19

similar finding as we found in the first experiment. Again, the ω-kA and the PSA produce more quality images when compared to other three algorithms. As the final comments, the BPA seems to have problems again when dealing with real-world data as it produces significant numerical noise as it can be seen from Figure 10(f).

Table 2 summarizes the performance of the presented five different algorithms in terms of processing time and the SCR over the conducted measurements. In assessing the speed of the algorithms, we observe similar performances as in the case of the simulation results that are listed in Table 1. Again, ω-kA is the fastest one and the BPA is the second best while the KMA is the worst. If we look at the SCR results, the clutter suppression ability of BPA is the best while the others demonstrate moderate performances when compared to BPA.

5. Conclusion and Discussions

In this paper, we have reviewed and compared fundamental algorithms that are used to focus the B-scan GPR data by presenting the algorithm steps. Namely, the hyperbolic summation, the Kirchhoff migration, the phase-shift migration, the frequency-wavenumber (ω-k) migration, and the back-projection based migration algorithms have been explained and applied to both the simulation and the measurement experiments. Based on the results obtained from the simulated and measured images, we have the following remarks for the investigated migration algorithms.

(i) The HSA is conceptually very simple and therefore it is very easy to implement. On the other hand, numerous hyperbolic template vectors should be applied for every pixel in the image which, in turn, is an expensive process in terms of the computation time. If the image size is relatively big, then the whole process of calculating the energy under the hyperbolic templates of every pixel may take a very long time. Although the focusing ability of the algorithm is not as good as the ω-kA and the BPA, the measurement results have showed that focusing success of the HSA also has some merit as it can be seen from the constructed GPR images. Also, the HSA produce fair contrast images when compared to others.

(ii) Although KM is conceptually similar to HSA, the algorithm is based on the scalar wave equation and it corrects the amplitude and the phases of some parameters such as spherical spreading and directivity that are not taken in to account in HSA. Therefore, the results obtained by KM are a little bit better than the ones with HSA that can also be seen from the resulted images. This is, of course, at the price of a longer simulation time. The improvement obtained by this amplitude correction can be clearly observed in Figures 7(c) and 10(c) to be compared to the images in Figures 7(b) and 10(b). The targets with smaller reflectivity are more visible in the images obtained with KM. By looking at both the simulation and the measurement results, KMA seems to provide the best resolution figures when compared to other four algorithms.

(iii) The PSA utilizes the ESM concept and iteratively tries to add phase-shifts to migrate the wave field to the exploding time of $t = 0$. The algorithm uses the advantage of FFT; therefore, it is much faster than the abovementioned algorithms. The PSA's performance in terms of computation time, focusing, and image quality is modest when compared to others.

(iv) Stolt migration algorithm (or the ω-kA) also makes use of the ESM concept and the scalar wave equation for the scattered field. It can be shown mathematically that the ω-kA provides the same solution set as in the case of KM. The ω-kA tries to constitute a 3D Fourier transform relationship between the image at the object space and the collected scattered field. Before applying the FFT routine, a mapping procedure from frequency-wavenumber domain to wavenumber-wavenumber domain is necessary. Although this mapping procedure may slow down the execution time of the algorithm, it is still fast thanks to the FFT step. This study suggests that the focusing performance and the image quality feature of the ω-kA are the best among all five algorithms. The algorithm is also fast when compared to HSA, KM, and BPA.

(v) As the BPA is not conceptually as simple as HSA or KM, the implementation of the algorithm is more complex. On the other hand, the BPA is much faster than these two algorithms since it takes and acts on the data as a block. Therefore, it requires less computation resources when compared to HSA or KM. This study has showed that the BPA is quite fast and its focusing ability is also very good. This can be viewed from both the simulated and the measured images and from the listed tables where BPA is the best for almost all the focusing parameters. One drawback of the BPA is due to its vulnerability to numerical noise. The algorithm produces comparably higher noise floor when compared to other four algorithms. This situation may cause resultant images of the low reflectivity targets to lie under the exaggerated noise floor level. Therefore, such targets may not be imaged within selection of dynamic range for the image display.

Conflict of Interests

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

Acknowledgment

This work is supported by the Scientific and Research Council of Turkey (TUBITAK) under Grant no. EEEAG-104E085.

References

[1] D. J. Daniels, *Surface-Penetrating Radar*, IEEE Press, 1996.

[2] L. Peters Jr., J. J. Daniels, and J. D. Young, "Ground penetrating radar as a subsurface environmental sensing tool," *Proceedings of the IEEE*, vol. 82, no. 12, pp. 1802–1822, 1994.

[3] S. Vitebskiy, L. Carin, M. A. Ressler, and F. H. Le, "Ultra-wideband, short-pulse ground-penetrating radar: simulation and measurement," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 35, no. 3, pp. 762–772, 1997.

[4] L. Carin, N. Geng, M. McClure, J. Sichina, and L. Nguyen, "Ultra-wide-band synthetic-aperture radar for mine-field detection," *IEEE Antennas and Propagation Magazine*, vol. 41, no. 1, pp. 18–33, 1999.

[5] W. A. Schneider, "Integral formulation for migration in two and three dimensions," *Geophysics*, vol. 43, no. 1, pp. 49–76, 1978.

[6] J. Gazdag, "Wave equation migration with the phase-shift method," *Geophysics*, vol. 43, no. 7, pp. 1342–1351, 1978.

[7] R. H. Stolt, "Migration by Fourier transform," *Geophysics*, vol. 43, no. 1, pp. 23–48, 1978.

[8] E. Baysal, D. D. Kosloff, and J. W. C. Sherwood, "Reverse time migration," *Geophysics*, vol. 48, no. 11, pp. 1514–1524, 1983.

[9] C. J. Leuschen and R. G. Plumb, "A matched-filter-based reverse-time migration algorithm for ground-penetrating radar data," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 39, no. 5, pp. 929–936, 2001.

[10] K. Gu, G. Wang, and J. Li, "Migration based SAR imaging for ground penetrating radar systems," *IEE Proceedings: Radar, Sonar and Navigation*, vol. 151, no. 5, pp. 317–325, 2004.

[11] J. Song, Q. H. Liu, P. Torrione, and L. Collins, "Two-dimensional and three-dimensional NUFFT migration method for landmine detection using ground-penetrating radar," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 44, no. 6, pp. 1462–1469, 2006.

[12] C. Cafforio, C. Prati, and F. Rocca, "Full resolution focusing of Seasat SAR images in the frequency-wave number domain," *International Journal of Remote Sensing*, vol. 12, no. 3, pp. 491–510, 1991.

[13] C. Cafforio, C. Prati, and F. Rocca, "SAR data focusing using seismic migration techniques," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 27, no. 2, pp. 194–207, 1991.

[14] A. S. Milman, "SAR imaging by ω-k migration," *International Journal of Remote Sensing*, vol. 14, no. 10, pp. 1965–1979, 1993.

[15] H. J. Callow, M. P. Hayes, and P. T. Gough, "Wavenumber domain reconstruction of SAR/SAS imagery using single transmitter and multiple/receiver geometry," *Electronics Letters*, vol. 38, no. 7, pp. 336–338, 2002.

[16] A. Gunawardena and D. Longstaff, "Wave equation formulation of synthetic aperture radar (SAR) algorithms in the time-space domain," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 36, no. 6, pp. 1995–1999, 1998.

[17] Z. Anxue, J. Yansheng, W. Wenbing, and W. Cheng, "Experimental studies on GPR velocity estimation and imaging method using migration in frequency-wavenumber domain," in *Proceedings of the 5th International Symposium on Antennas, Propagation and EM Theory (ISAPE '00)*, pp. 468–473, Beijing, China, 2000.

[18] C. Gilmore, I. Jeffrey, and J. LoVetri, "Derivation and comparison of SAR and frequency-wavenumber migration within a common inverse scalar wave problem formulation," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 44, no. 6, pp. 1454–1461, 2006.

[19] J. Gazdag and P. Sguazzero, "Migration of seismic data," *Proceedings of the IEEE*, vol. 72, no. 10, pp. 1302–1315, 1984.

[20] C. Ozdemir, S. Demirci, and E. Yigit, "A review on the migration methods in B-scan ground penetrating radar imaging," in *Proceedings of the Progress in Electromagnetics Research Symposium (PIERS '12)*, pp. 789–793, Kuala Lumpur, Malaysia, March 2012.

[21] J. F. Claerbout, *Imaging the Earth's Interior*, Blackwell Scientific Publications, 1985.

[22] C. Ozdemir, S. Demirci, E. Yigit, and A. Kavak, "A hyperbolic summation method to focus B-scan ground penetrating radar images: an experimental study with a stepped frequency system," *Microwave and Optical Technology Letters*, vol. 49, no. 3, pp. 671–676, 2007.

[23] O. Yilmaz, *Seismic Data Processing*, Society of Exploration Geophysicists, 1987.

[24] D. S. Jones, *The Theory of Electromagnetism*, Pergamon Press, New York, NY, USA, 1964.

[25] P. M. Morse and H. Feshbach, *Methods of Theoretical Physics*, McGraw-Hill, New York, NY, USA, 1953.

[26] I. Lecomte, S.-E. Hamran, and L.-J. Gelius, "Improving Kirchhoff migration with repeated local plane-wave imaging? A SAR-inspired signal-processing approach in prestack depth imaging," *Geophysical Prospecting*, vol. 53, no. 6, pp. 767–785, 2005.

[27] J. A. Stratton, *Electromagnetic Theory*, McGraw-Hill, New York, NY, USA, 1941.

[28] D. C. Munson Jr., J. D. O'Brien, and W. K. Jenkins, "A tomographic formulation of spotlight-mode synthetic aperture radar," *Proceedings of the IEEE*, vol. 17, no. 8, pp. 917–925, 1983.

[29] J. L. Bauck and W. K. Jenkins, "Tomographic processing of spotlight-mode synthetic aperture radar signals with compensation for wavefront curvature," in *Proceedings of the International Conference on Acoustics, Speech, and Signal Processing (ICASSP '88)*, vol. 2, pp. 1192–1195, New York, NY, USA, 1988.

[30] R. Mersereau and A. Oppenheim, "Digital reconstruction of multidimensional signals from their projections," *Proceedings of the IEEE*, vol. 62, no. 10, pp. 1319–1338, 1974.

[31] E. Fisher, G. A. McMechan, A. P. Annan, and S. W. Cosway, "Examples of reverse-time migration of single-channel, ground- penetrating radar profiles," *Geophysics*, vol. 57, no. 4, pp. 577–586, 1992.

[32] L. Capineri, P. Grande, and J. A. G. Temple, "Advanced image-processing technique for real-time interpretation of ground-penetrating radar images," *International Journal of Imaging Systems and Technology*, vol. 9, no. 1, pp. 51–59, 1998.

[33] I. L. Morrow and P. A. van Genderen, "2D polarimetric backpropagation algorithm for ground-penetrating radar applications," *Microwave and Optical Technology Letters*, vol. 28, pp. 1–4, 2001.

[34] C. Ozdemir, *Inverse Synthetic Aperture Radar Imaging with Matlab Algorithms*, John Wiley & Sons, Hoboken, NJ, USA, 2012.

[35] A. Martinez and J. L. Marchand, "SAR image quality assessment," *Revista de Teledetección*, no. 2, pp. 1–7, 1993.

[36] E. Yigit, S. Demirci, C. Ozdemir, and M. Tekbas, "Short-range ground-based synthetic aperture radar imaging: performance comparison between frequency-wavenumber migration and back-projection algorithms," *Journal of Applied Remote Sensing*, vol. 7, 2013.

Hindawi Publishing Corporation
Mathematical Problems in Engineering
Volume 2014, Article ID 871859, 11 pages
http://dx.doi.org/10.1155/2014/871859

Research Article

Analyses of Effects of Cutting Parameters on Cutting Edge Temperature Using Inverse Heat Conduction Technique

Marcelo Ribeiro dos Santos,[1] Sandro Metrevelle Marcondes de Lima e Silva,[2] Álisson Rocha Machado,[1] Márcio Bacci da Silva,[1] Gilmar Guimarães,[1] and Solidônio Rodrigues de Carvalho[1]

[1] *College of Mechanical Engineering, Federal University of Uberlândia, Campus Santa Mônica, Bloco M, Avenida João Naves de Ávila 2121, 38408-100 Uberlândia, MG, Brazil*
[2] *Institute of Mechanical Engineering, Federal University of Itajubá, Campus Professor José Rodrigues Seabra, Avenida BPS 1303, 37500-903 Itajubá, MG, Brazil*

Correspondence should be addressed to Solidônio Rodrigues de Carvalho; srcarvalho@mecanica.ufu.br

Received 17 February 2014; Revised 2 April 2014; Accepted 23 April 2014; Published 1 June 2014

Academic Editor: Caner Özdemir

During machining energy is transformed into heat due to plastic deformation of the workpiece surface and friction between tool and workpiece. High temperatures are generated in the region of the cutting edge, which have a very important influence on wear rate of the cutting tool and on tool life. This work proposes the estimation of heat flux at the chip-tool interface using inverse techniques. Factors which influence the temperature distribution at the AISI M32C high speed steel tool rake face during machining of a ABNT 12L14 steel workpiece were also investigated. The temperature distribution was predicted using finite volume elements. A transient 3D numerical code using irregular and nonstaggered mesh was developed to solve the nonlinear heat diffusion equation. To validate the software, experimental tests were made. The inverse problem was solved using the function specification method. Heat fluxes at the tool-workpiece interface were estimated using inverse problems techniques and experimental temperatures. Tests were performed to study the effect of cutting parameters on cutting edge temperature. The results were compared with those of the tool-work thermocouple technique and a fair agreement was obtained.

1. Introduction

Nowadays, several researchers proposed the combination of inverse techniques and analytical or numerical heat transfer solutions to analyze the thermal fields during machining processes. One method to study the heat transfer problem is to adopt a known heat flux and calculate the temperature at the chip-tool interface from the solution of the heat diffusion equation. In literature, this methodology is called direct problem in heat transfer. However, during machining the experimental heat flux is unknown and inverse techniques have to be used to predict this parameter. This proposal estimates the transient heat flux with a numerical model based on the heat diffusion equation using inverse techniques and experimental temperatures measured at accessible regions of the sample.

In Chen et al. [1] the inverse technique used was based on the sequential function specification method proposed by Beck et al. [2]. The numerical technique used was the finite volume method and the temperatures were obtained by inserting a thermocouple in the tool. In both methods the model took into account only the insert and some discrepancies were observed between calculated and measured temperatures.

Lazard and Corvisier [3] considered the problem of estimating the transient temperature and the heat flux at the chip-tool interface during a turning process using an inverse approach. The heat transfer model was based on a quadrupole formulation commonly used to solve ordinary differential equations in the Laplace domain. The results of temperature obtained with the analytical model used were

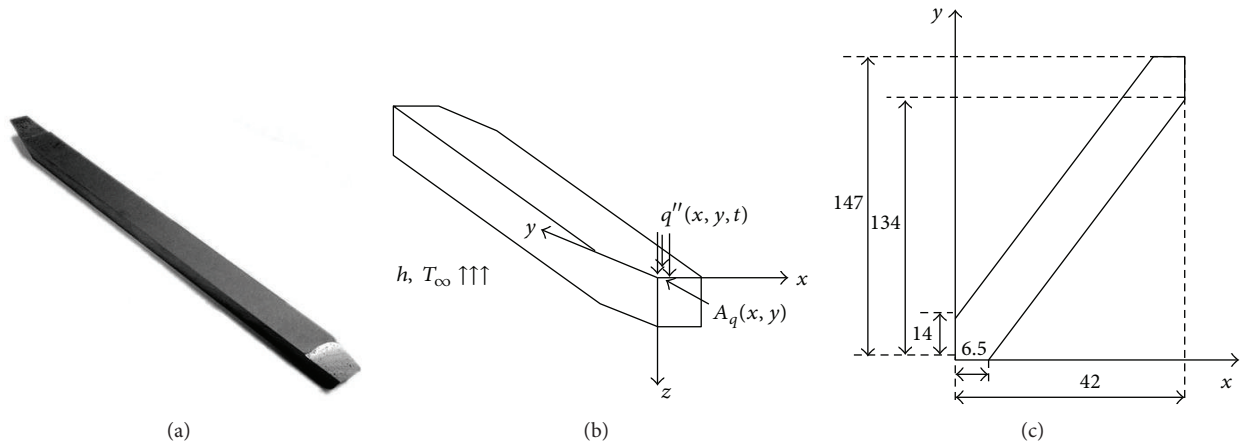


Figure 1: (a) High speed steel tool, (b) three-dimensional physical model, and (c) tool dimensions in millimeters (mm) where the coordinate "z" is 9.5 (mm).

in good agreement with those obtained with FLUENT. The authors assumed that the temperature measurements were available for the inverse analysis.

In the work of Yvonnet et al. [4] an innovative approach was proposed. The work was based on a simple inverse procedure to identify both the heat flux flowing into the tool through the rake face and the heat transfer coefficient between the tool and the environment during a typical orthogonal cutting process. To determine the heat flux and convection heat transfer coefficient, an iterative Newton-Raphson procedure was used to minimize the error defined by the difference between experimental and calculated temperatures. Four different tools were used; three of them were manufactured by electrical discharge machining (EDM) cutting the last slot with different distances between the tool tip and the slot: 0.35, 0.50, and 0.60 mm. The last tool was instrumented with a thermocouple to measure the average heat flux flowing into the tool through the rake face.

Woodbury et al. [5] demonstrated the solution of a three-dimensional inverse heat conduction problem using an evolutionary algorithm (EA). The heat flux from the workpiece into the tool during the turning process was determined using evolutionary operations combined with measurements of surface temperature on the tool. The three-dimensional conduction in the tool and tool holder was simulated using FLUENT.

A numerical prediction of the three-dimensional temperature fields in moving chip, stationary tool, and moving workpiece during machining operations using a finite difference method was presented in Ulutan et al. [6]. First, the authors studied the chip and tool together to create a heat balance due to the friction on the rake face. Then, an investigation of the effect of the chip and tool interface temperature on the temperature field of the workpiece, taking into account the heat generated at the shear plane, was made. A finite difference model was employed to describe the process, and the results were in good agreement with experimental data presented in the literature. The advantage of this method in relation to finite element method (FEM) models is the computational time, which decreased substantially, and the advantage in relation to curve fitting methods is that the method is based on the physical phenomenon.

In Samadi et al. [7] the sequential function specification method was used with simulated temperature data to estimate the transient heat flux imposed on the rake face of a cutting tool during the cutting operation with two different hypotheses. The thermal conductivity was considered constant in one, and in the other it varied with temperature. The cutting tool was modeled as a three-dimensional object. The influence of nonlinearity and different sensor locations were investigated in order to determine an optimal experimental procedure. Finally, typical temperature data during turning was used to recover the heat flux at the cutting tool surface considering linear and nonlinear solutions.

The aim of the present paper is to estimate the heat flux and calculate the temperature field at the cutting interface of the high speed steel tool. In addition, the influence of the cutting parameters (cutting speed, cutting depth, and feed rate) on the tool-chip interface temperature was also investigated. The novelty in this work, compared to others presented in the literature, is the joint solution and analysis of the following problems: a nonlinear tridimensional thermal model, an inverse technique to estimate the heat flux, the thermal analyses of a machining process, and the final comparison of the results with those obtained from a fully experimental methodology proposed to Evangelista Luiz [8] based on the tool-workpiece thermocouple technique (TWTT).

2. The Direct Problem: Thermal Model

Figure 1 shows the high speed steel tool, the model used, and the dimensions of the tool. The interface contact area $A_q(x, y)$ was subjected to the heat flux $q''(x, y, t)$ generated by contact between the tool and the workpiece. At the remaining boundaries a constant convective heat transfer coefficient of 20 (W/m^2 K) was considered.

The three-dimensional physical problem was solved in Cartesian coordinates, using finite volume technique with

Table 1: Thermal properties of high speed steel tool according to Taylor Specials Steels Ltda (2009). Material: AISI M32 C, with 10% of cobalt.

Temperature range (°C)	$0 \le T \le 400$	$T > 400$
Thermal conductivity (W/mK)	$0.0105\,T + 23.8$	$-0.005\,T + 30$
Thermal diffusivity (m^2/s)	$-5.03 \times 10^{-10}\,T + 7.02 \cdot 10^{-6}$	$-5.94 \times 10^{-9} \cdot T + 9.19 \times 10^{-6}$

irregular mesh. The objective is to obtain the temperature distribution in the tool using the direct problem and in following estimate the heat flux generated at the chip-tool interface with inverse techniques.

The choice of the high speed steel tool is due to its large application in industry and also because Evangelista Luiz [8] presented a thermal analysis using tool/work thermocouple technique (TWTT), thus permitting the comparison with the results obtained using a different technique.

The thermal problem shown in Figure 1(b) is described by the heat diffusion equation as in Carvalho et al. [9]:

$$\frac{\partial}{\partial x}\left(\lambda\frac{\partial T}{\partial x}\right) + \frac{\partial}{\partial y}\left(\lambda\frac{\partial T}{\partial y}\right) + \frac{\partial}{\partial z}\left(\lambda\frac{\partial T}{\partial z}\right) = \rho C\frac{\partial T}{\partial t}. \quad (1)$$

The boundary conditions imposed are

$$-\lambda\frac{\partial T}{\partial \eta} = h\left(T - T_\infty\right) \quad (2)$$

at the regions exposed to the environment and

$$-\lambda\frac{\partial T}{\partial \eta} = q''\left(x, y, t\right) \quad (3)$$

at the interface defined by A_q, where η is the outward normal in coordinates x, y, and z; T is the temperature; T_∞ is the room temperature; λ is the thermal conductivity; ρC is the volumetric heat capacity; and h the heat transfer coefficient. The initial condition is given by

$$T\left(x, y, z, 0\right) = T_o, \quad (4)$$

where T_o is the initial temperature of the tool, shim, and tool holder.

The above equations were implemented in C++. An algorithm called Inverse3D or Inv3D was developed. This computational algorithm has been developed over the last 14 years by the research group of the Laboratory of Heat and Mass Transfer, College of Mechanical Engineering, Federal University of Uberlândia. It has been extensively validated and has led to several articles published in scientific journals and conferences, such as Brito et al. [10], Brito et al. [11], Sousa et al. [12] and Carvalho et al. [9].

The thermal properties of the tool obtained from Taylor Special Steels Ltd. [13] are shown in Table 1.

Several factors are responsible for the errors or uncertainties in the mathematical model. Comparing Figures 1(a) and 1(b), there are several simplifications in geometry of the thermal model compared to the cutting tool. The model did not include the rake and relief angles typical of any cutting tool. Such simplifications imply an increase in volume of the model of 1.47% compared to the tool. In spite of this small increase, no significant changes were identified in the thermal field calculated by the model of the cutting tool.

As for the uncertainties in the thermal model, the thermal properties of the tool were obtained from scientific papers. Other sources of uncertainties are the correct location of the thermocouples and the presence of noise in the experimental signal of the temperature. In this work, we opted for the technique of capacitive discharge to weld the thermocouples to the tool, thus minimizing uncertainty regarding the thermal contact resistance.

An HP 75000, series B, data acquisition system with E1326B voltmeter controlled by a PC, connected to a stabilized power source, was used. The temperature acquisition interval was 0.25 seconds. Based on the procedures and the equipment adopted, it was estimated that the measurement error of the entire system (thermocouple/multimeter) was less than ±0.3°C.

Also an average h of 20 W/m^2 K for the entire surface of the tool was used. The correct determination of the heat transfer coefficient by convection is not an easy task. An analysis of the influence of this important parameter on the calculated temperature at the cutting interface is shown in Figure 2.

Figure 2(a) shows that the heat transfer coefficients by convection (h) in the range tested (10–30 W/m^2 K) has little influence on the final temperature at the cutting interface. Adopting an average h of 20 W/m^2 K as reference and comparing the calculated temperature with those obtained for other values of h a maximum deviation of less than 0.74% was obtained during machining as shown in Figure 2(b). With this result, it was concluded that for the range of values of h considered, the calculation of the temperature at the cutting interface was not compromised. Hence the value of 20 W/m^2 K was adopted.

Figure 3 shows the heat transfer rate by convection at the cutting interface for different values of h.

Figure 3 shows that as h increases, the rate of heat transfer becomes more significant. Besides, within the range of values analyzed (10–30 W/m^2 K) a maximum difference of only 1 W in relation to h = 20 W/m^2 K occurred. Thus, it is concluded that h has little influence on the interface temperature. The energy lost by convection represents 6.1% of the energy generated at the cutting interface (h = 20 W/m^2 K).

3. The Inverse Problem: Function Specification Procedure

The inverse technique adopted in this work is the Sequential Function Estimation, Beck's Method [2]. This technique requires the sensitivity coefficients (ϕ), which are the derivatives of the calculated temperatures (T) with respect to the heat flux (q'').

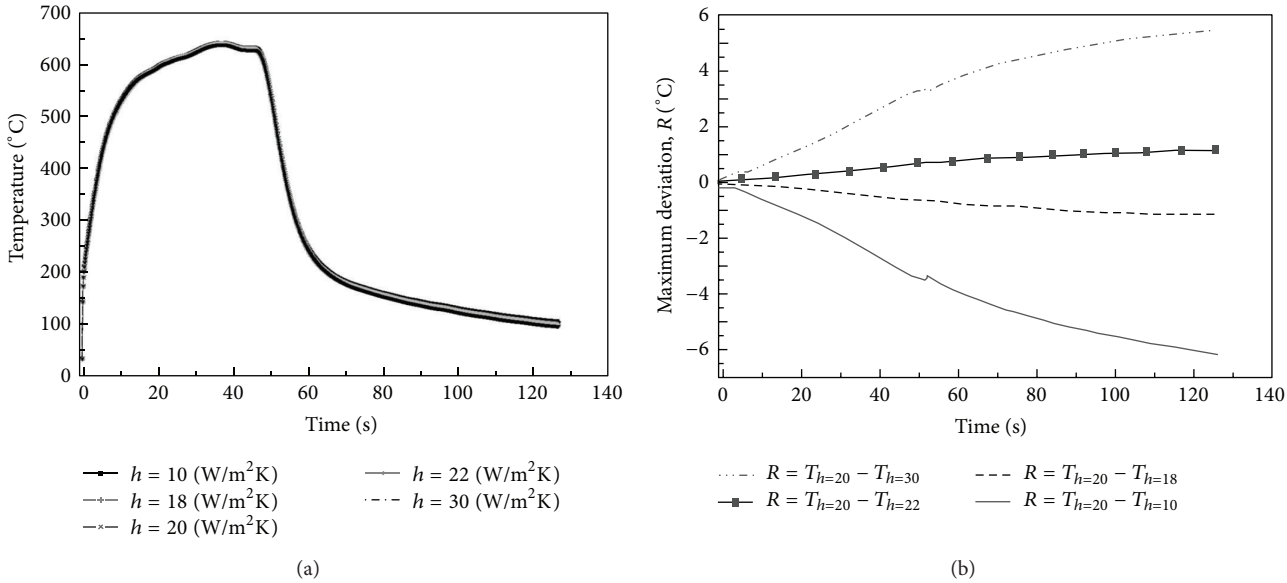


Figure 2: Analysis of the influence of the heat transfer coefficient by convection: (a) cutting interface temperature for different values of h and (b) analysis of uncertainty considering $h = 20\ \mathrm{W/m^2\ K}$ as reference. Cutting conditions: feed rate: 0.138 mm/rot, cutting speed: 142 m/min (900 rpm), and depth of cut: 1.0 mm.

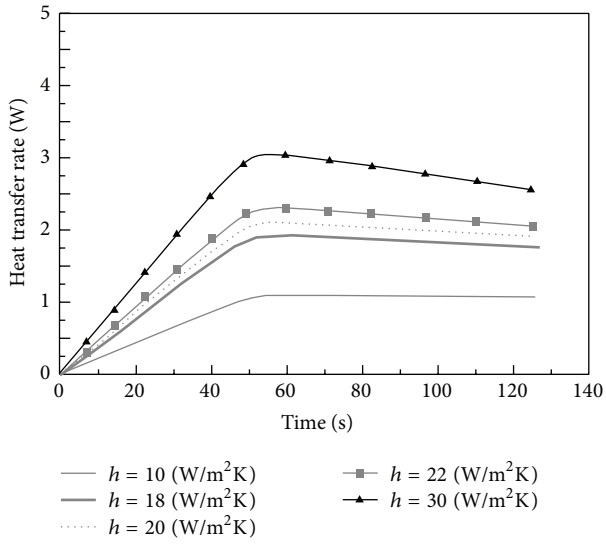


Figure 3: Heat transfer rate by convection for different values of h. Cutting conditions: feed rate: 0.138 mm/rot, cutting speed: 142 m/min (900 rpm), and depth of cut: 1.0 mm.

Numerically, the sensitivity coefficients (ϕ) were calculated at the thermocouples positions (Table 2) as shown in Figure 4(a). The direct problem is solved considering (1), with $q'' = 1$ (unit value), initial temperature equal to zero ($T = 0$), h equal to $20\ \mathrm{W/m^2\ K}$, and constant thermal properties calculated for $T = 30^{\circ}\mathrm{C}$ using Table 1. The numerical technique is based on Duhamel's summation and Stolz method [2].

Basically, the Sequential Function Estimation minimizes a squared error function based on numerical (T) and experimental temperatures (Y), to estimate the heat flux (q'') at

Table 2: Thermocouple locations following the coordinate system defined in Figure 1.

Position	Thermocouple				
	1	2	3	4	5
x (mm)	0.61	2.70	0.0	3.30	2.00
y (mm)	7.20	8.50	9.0	7.00	3.40
z (mm)	0.0	0.0	5.0	9.50	9.50

chip-tool contact area for each time step (M). The basic idea is to reduce a continuous function to a set of parameters by specifying an underlying nature of the function. In this work a constant sequential function was adopted. Thus, according to Beck et al. [2], the heat flux can be estimated as given in

$$q''_M = \frac{\sum_{i=1}^{r} \sum_{j=1}^{J} \left(Y_{j,M+i-1} - \hat{T}_{j,M+i-1|q_{M=\cdots=0}}\right) \phi_{ji}}{\sum_{i=1}^{r} \sum_{j=1}^{J} {\phi_{ji}}^2}, \quad (5)$$

where M is the time, J is the number of sensors, r is the number of future time steps, and Y and T are the experimental and calculated temperatures, respectively.

In the estimation process 10 future time steps were used for each cutting condition. The transient heat flux for each experiment was estimated as shown in Figure 4(b).

Finally, according to Figure 4(c) the direct problem is again solved considering the nonlinear thermal model (1) in which the thermal properties vary with temperature, resulting in the temperature distribution in the cutting tool.

4. Experimental Procedure

The machining test was carried out in a conventional IMOR MAXI-II-520-6CV lathe without coolant. The material used

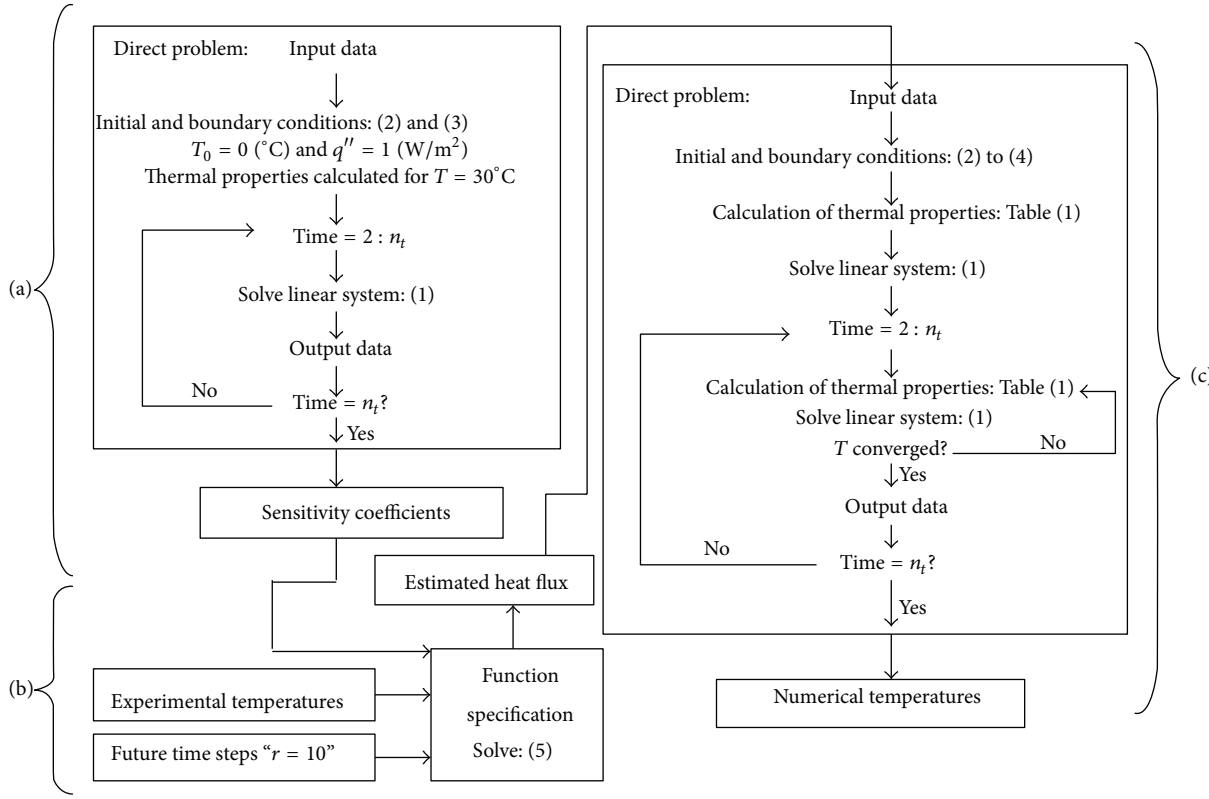
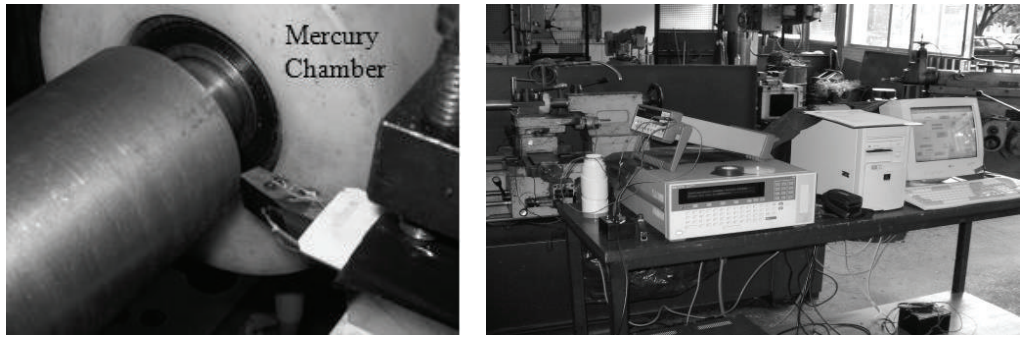


Figure 4: Computational algorithm: (a) calculation of the sensitivity coefficients; (b) Beck's method for the solution of the inverse problem; (c) solution of the nonlinear thermal model.



(a) (b)

Figure 5: (a) Thermocouples on the tool and (b) equipment.

in the experimental tests was cylindrical ABNT 12L14 [8] bars with an external diameter of 50.2 mm. Figure 5 shows the temperatures measured at accessible locations of the insert using type T thermocouples and an HP 75000, series B, data acquisition system with an E1326B voltmeter controlled by a PC.

Table 2 gives the thermocouple locations according to the coordinate system shown in Figure 1. To evaluate the influence of the machining parameters—depth of cut (ap), feed rate (f), and cutting speed (Vc)—on temperature at chip-tool interface, the following tests given in Tables 3, 4, and 5 were made.

The chip-tool contact area of each test was defined by an imaging system. The device consists of a Hitachi model KP-110 CCD video camera, an AMD K6 450 MHz computer, and a software processer of images (the Global Lab Image). Figure 6 shows a photograph of the contact area ($A_q = L \cdot H$) for cutting conditions: $ap = 1.0$ mm, $f = 0.138$ mm/rot, and $Vc = 56$ m/min (scale 25 : 1).

Figure 7 shows the results of the chip-tool contact areas for the cutting conditions of Tables 3 to 5.

Table 3: Depth of cut (constant parameters: $f = 0.138$ mm/rot and $Vc = 56$ m/min).

	Unit (mm)			
ap	0.5	1.0	1.5	2.0

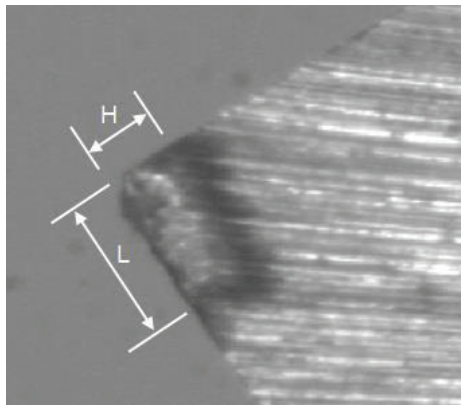


Figure 6: Chip-tool contact area for cutting conditions: $ap = 1.0$ mm; $f = 0.138$ mm/rot and; $Vc = 56$ m/min; scale 25 : 1.

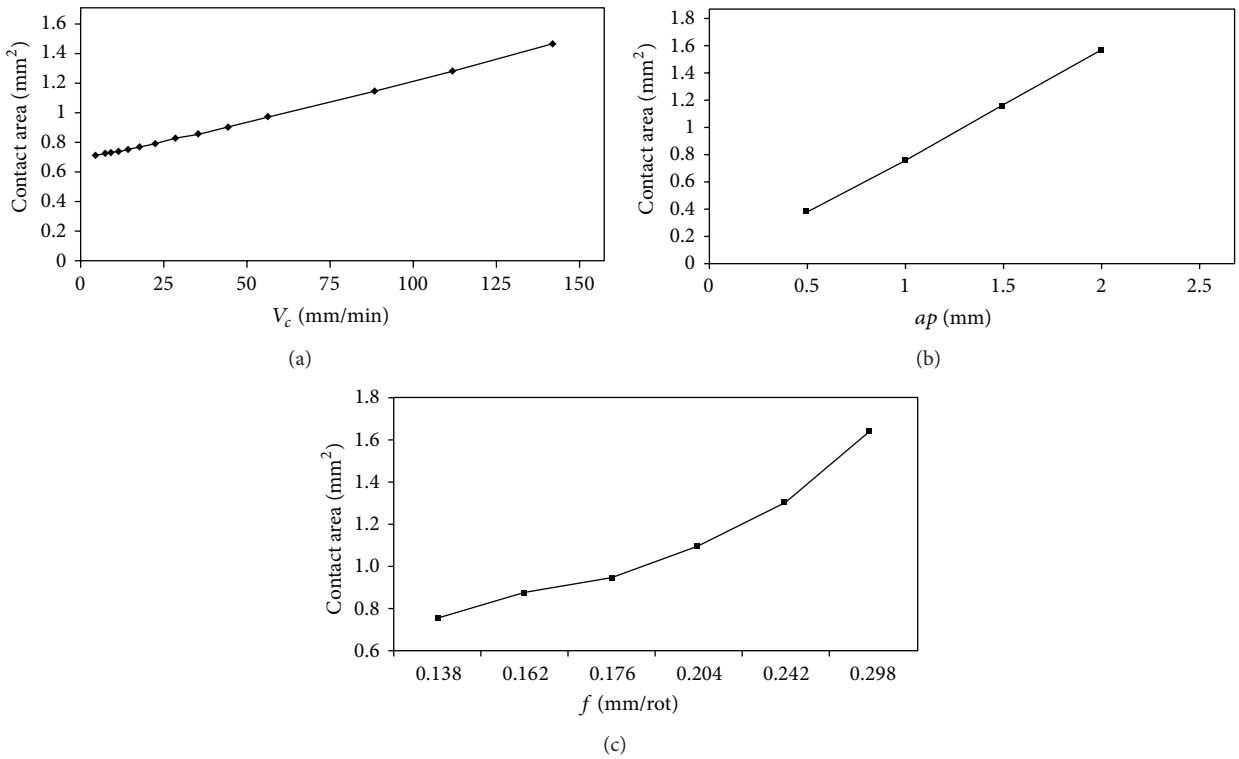


Figure 7: Chip-tool contact area for the following conditions: (a) $ap = 1.0$ mm, $f = 0.138$ mm/rot, and Vc variable; (b) $f = 0.138$ mm/rot, $Vc = 56$ m/min, and ap variable; (c) $ap = 1.0$ mm, $Vc = 56$ m/min, and f variable.

Table 4: Feed rate (constant parameters: $ap = 1.0$ mm and $Vc = 56$ m/min).

	Unit (mm/rot)					
f	0.138	0.162	0.176	0.204	0.242	0.298

Increase of the cutting parameters increases the chip-tool contact area. The development of an accurate method to measure the chip-tool contact area represents a great challenge because direct observations during cutting are not possible. Most of the available theories of the identification of the chip-tool contact area are derived from the study of the interface after cutting had been interrupted. In this work, the methodology is also based on the analysis of the contact area after cutting. However, even with the enlarged areas from analysis software, identification of the contact area is not an easy task and requires experience and knowledge of the researcher. Normally approximate areas are obtained because even with the various existing theories it is difficult or even impossible to identify the real chip-tool contact area.

In this work, the area was approximated as a rectangle, in which the energy is distributed evenly. Thus,

$$q''(x, y, t) = q''(t). \quad (6)$$

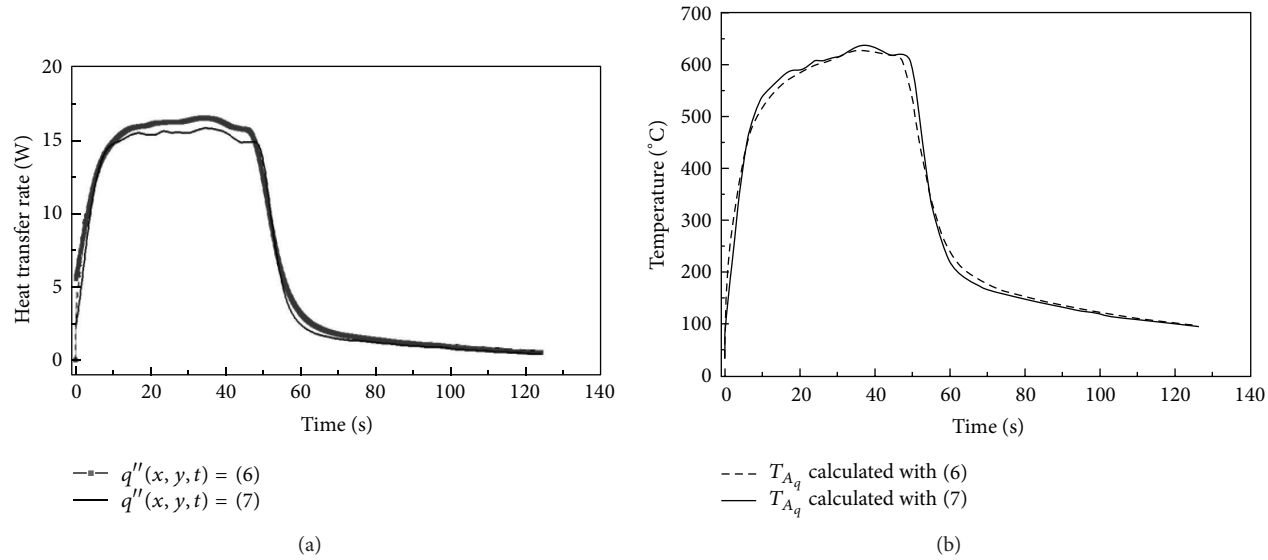


Figure 8: (a) Comparison of the average heat transfer rates and (b) temperatures at the cutting interface.

Table 5: Cutting speed (constant parameters: ap = 1.0 mm and f = 0.138 mm/rot).

	Unit (m/min)													
Vc	4.4	7.1	8.8	11.2	14.2	17.7	22.1	28.4	35.3	44.2	56	88.3	112	142

However, metallographic techniques such as those applied by Dearnley [14] showed that the correct methodology would be to adopt an irregular area and a nonuniform heat flux distribution at the cutting interface. However, the combination of both parameters in the thermal model, that is, irregular heat flux and irregular area, would lead to countless possible results. Thus, as the area had been previously measured, it was decided to change only the distribution of the heat flux at the cutting interface. The thermal model was simulated in two stages: initially a uniform heat flux at the cutting interface was adopted (6), and in the following stage, an exponential heat flux was used, as in

$$q''(x, y, t) = q''_a \cdot e^{(-1/l_x^2)\cdot(x-x_o)^2+(-1/l_y^2)\cdot(y-y_o)^2}. \quad (7)$$

The variables x and y define the coordinates of the contact area, q''_a is the maximum value of instantaneous heat flux at the contact area, l_x and l_y are the dimensions of the contact area, and x_0 and y_0 are the points of maximum value. The variables l_x, l_y, x_o, and y_o were adjusted for each cutting condition, using the contact areas shown in Figure 7.

Thus, with the cutting parameters, the chip-tool contact area, and the experimental temperatures measured for each case, this work proposes to solve the inverse problem, estimate the heat flux at the interface, and obtain the three-dimensional temperature distribution in the tool. The results were compared with those obtained by the experimental tool-work thermocouple method [8].

5. Results and Discussions

Figure 8 shows the heat transfer rate and the average temperature at the cutting interface during the most severe cutting condition: feed rate f = 0.138 mm/rot, cutting speed Vc = 142 m/min (900 rpm), and cutting depth ap = 1.0 mm.

Figure 8(a) shows that the average heat transfer rate estimated with an exponential heat flux (7) is quite similar to that estimated with a uniform heat flux (6). Consequently the average temperatures at the cutting interface are quite similar (Figure 8(b)).

Figure 9 shows a comparison of experimental and calculated temperatures of each methodology.

Figure 9(a) shows good agreement between calculated and experimental temperatures. However, Figure 9(b) shows that the residue is lower for temperatures calculated with exponential heat flux (7).

Figure 10 shows the temperature profile in the tool for the following cutting conditions: feed rate f = 0.138 mm/rot, cutting speed Vc = 142 m/min (900 rpm), and cutting depth ap = 1.0 mm, where the maximum temperature calculated at the chip-tool interface was 600°C.

Analyzing Figures 10(b) and 10(c) the isotherms in a real machining process tend to behave as shown in Figure 10(c); that is, the maximum temperature in the contact area is at a certain distance from the main cutting edge as found in the literature, such as Dearnley [14]. Thus the use of an exponential heat flux gives results more consistent with those identified in a real turning process.

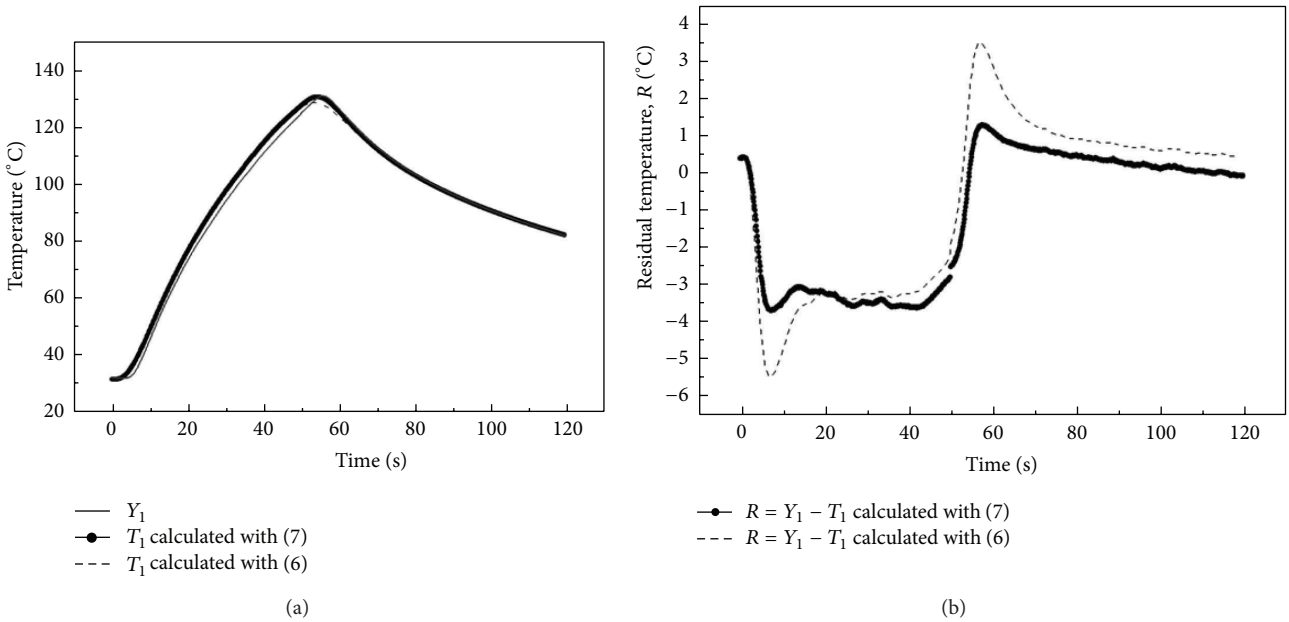


Figure 9: (a) Comparison between experimental and calculated temperatures and (b) residual temperatures.

Figure 11 shows the average maximum temperature at the cutting interface for the cutting conditions defined in Tables 3 to 5. These values were calculated using the numerical methodology proposed in this work (considering (6) and (7), uniform and exponential heat flux) and measured with the experimental technique TWTT [8].

As in Trent [15], the temperature at the cutting interface increases with the increase in cutting parameters. In other words, increasing the cutting parameters increases the chip-tool contact area and the rate of deformation, thus generating more energy and consequently higher temperatures at the cutting interface.

Figure 11 also shows the difference between temperature profiles measured with TWTT and calculated by numerical techniques. With the experimental TWTT technique as a reference, there is a significant difference between experimental and numerical temperatures. Figure 11(c) shows that this difference decreases when the cutting speed increases. Trent [15] showed that cutting speeds between 100 and 200 m/min give temperatures at cutting interface between 600°C and 800°C, which agrees with the values obtained in this work.

Both TWTT and numerical methodology have sources of errors which can influence directly the measured and calculated temperatures which can justify the differences shown in Figure 11.

For the numerical methodology the uncertainties are related to the dimensions and the simplifications adopted to simulate the thermal behavior of tool, the correct identification of the thermal properties of the high speed steel, the convective heat transfer coefficient, the measured temperatures during turning, and the correct identification of the chip-tool contact area for each cutting condition.

According to Evangelista Luiz [8] the uncertainties in TWTT technique are related to the calibration of the system and the assembly of the experimental components. The TWTT basically consists of an electric circuit which involves the tool, the workpiece, and the lathe. The tool and the workpiece forming the thermocouple had to be previously calibrated as any conventional sensor and the experimental procedure is strongly dependent on the equipments and the ability of the operator. Besides, during turning, a mercury chamber had to be used to connect the components of the system due to the rotation of the workpiece as shown in Figure 5(a). Also the attrition between tool and workpiece generates noise in the signal of the temperature that had to be minimized using statistical tools to remove excessive noise and to define the mean temperature and the standard deviation.

6. Conclusions

This work is an interdisciplinary study involving two major areas of mechanical engineering: heat transfer and manufacturing processes. Based on the knowledge of these two areas, a new computational algorithm for solving problems of heat transfer applied to manufacturing processes, focusing on turning process, was developed.

The simulations optimize the numerical model proposed analyzing the numerical mesh, computational cost, convergence, quality of numerical results, and possible sources of error in order to obtain a favorable cost-benefit solution of the thermal problem. Furthermore, in experiments in this study, it was sought to minimize the likely sources of experimental errors.

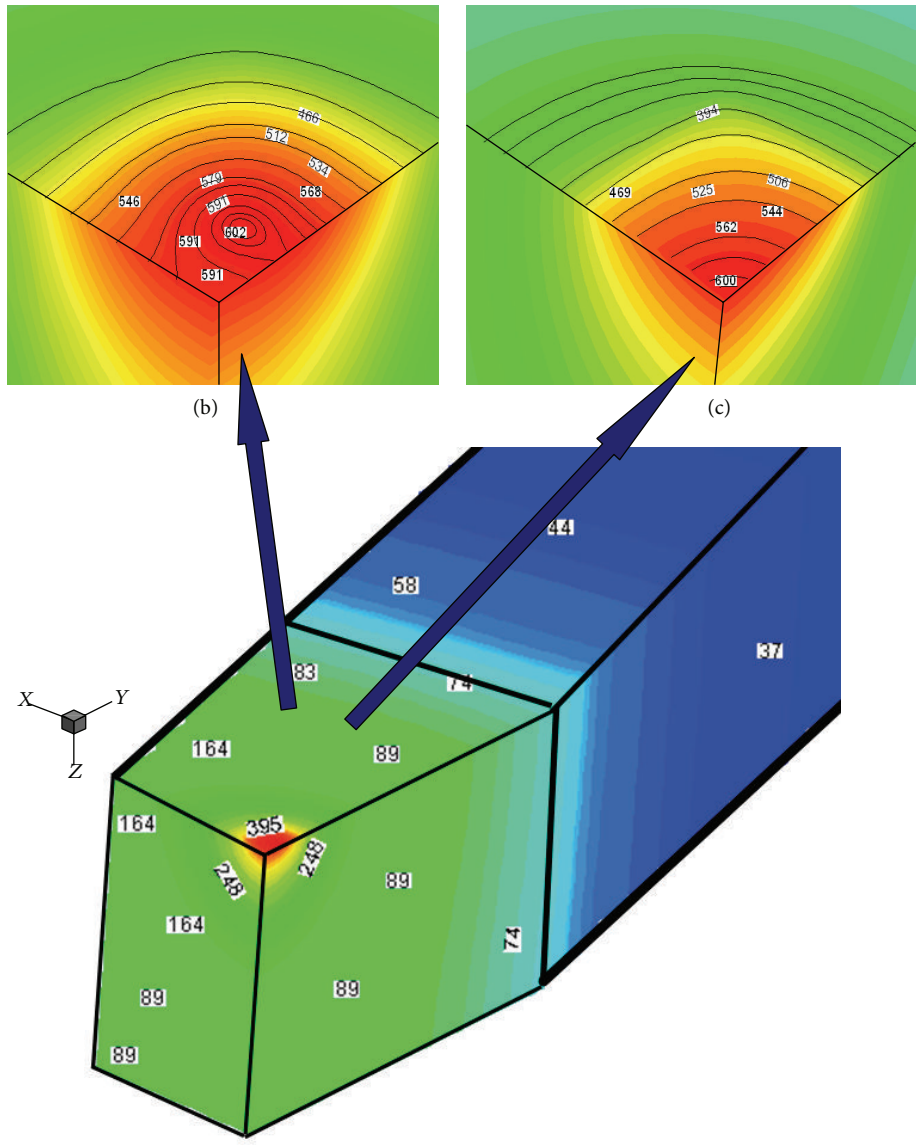


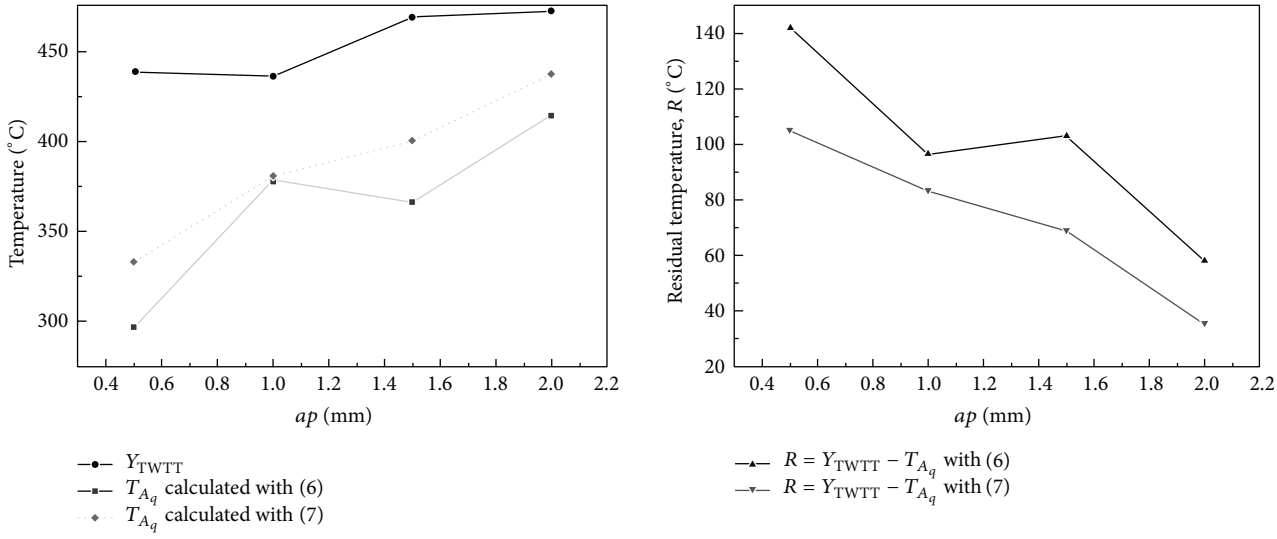
FIGURE 10: Temperature profile for feed rate $f = 0.138$ mm/rot, cutting speed $Vc = 142$ m/min (900 rpm), and cutting depth $ap = 1.0$ mm: (a) tool, (b) cutting interface temperatures for exponential heat flux (7), and (c) cutting interface temperatures for uniform heat flux (6).

Regarding the thermal effects studied, it was possible to calculate and analyze the three-dimensional temperature distribution in the thermal machining model as well as at chip-tool interface. The numerically temperatures were compared with experimental data which can increase the reliability and credibility of the results found. Besides temperature, in this study the heat flux at the contact interface, which allowed a quantitative analysis of the thermal energy generated in the machining process, was estimated.

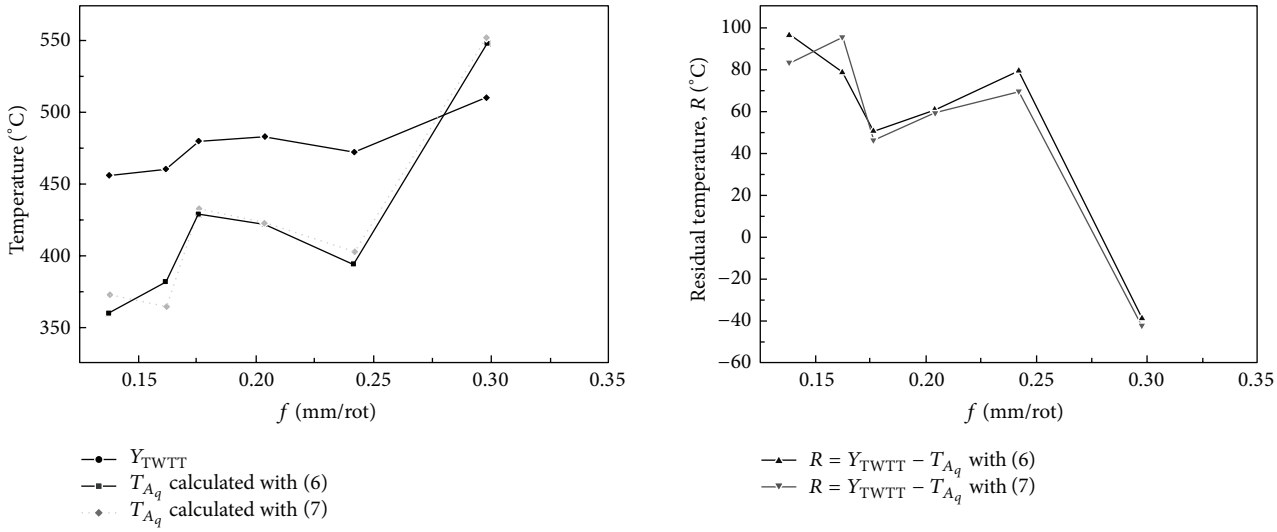
This work studied the thermal effects during turning and also analyzes the influence of the cutting conditions (cutting speed, feed rate, and depth of cut) on the temperature generated at chip-tool interface. Analyzing the results, as in the literature, the temperature at the cutting interface increases with the increase in cutting conditions. The results do not fit perfectly with those presented by Evangelista Luiz [8], using the experimental method of tool-workpiece thermocouple technique. This fact is attributed to the possible sources of error in each methodology which have direct influence on the results. Thus, there is no existing technique that can be universally accepted as a standard. There are attempts to understand the fundamentals of the thermal exchanges during turning and it is believed that the understanding is the next step to predict the performance of a manufacturing process [16].

Conflict of Interests

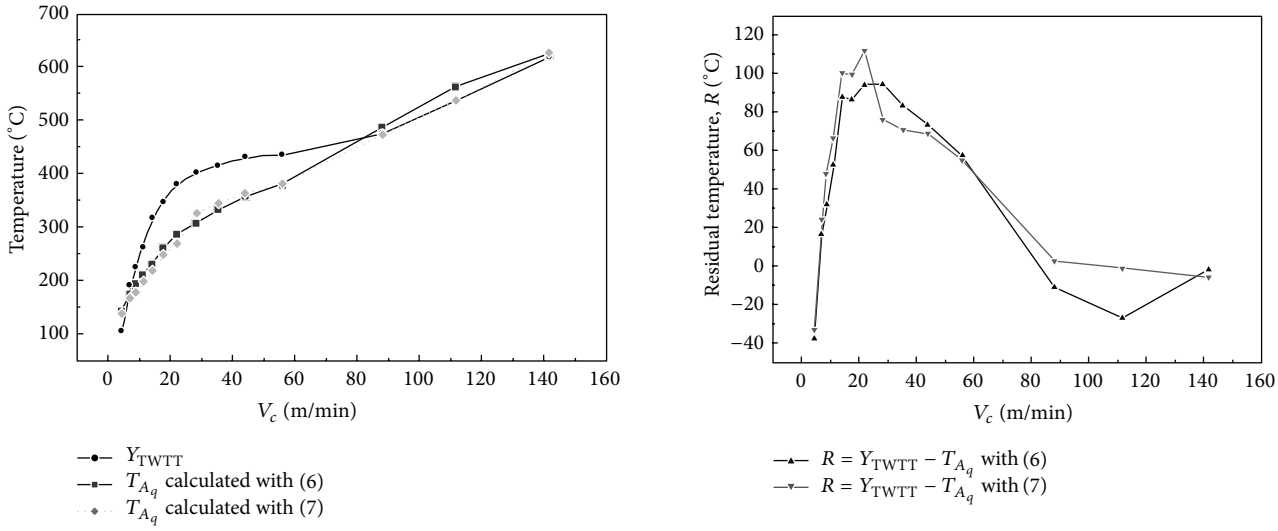
The authors declare that there is no conflict of interests regarding the publication of this paper.



(a)



(b)



(c)

Figure 11: Chip-tool interface temperature and residual for TWTT and data from (a) Table 3, (b) Table 4, and (c) Table 5.

Acknowledgments

The authors thank CNPq, FAPEMIG, and CAPES for the financial support, without which this work would not be possible.

References

[1] W. C. Chen, C. C. Tsao, and P. W. Liang, "Determination of temperature distributions on the rake face of cutting tools using a remote method," *International Communications Heat Mass Transfer*, vol. 24, pp. 161–170, 1997.

[2] J. V. Beck, B. Blackwell, and C. St. Clair, *Inverse Heat Conduction: Ill-Posed Problems*, Wiley-Interscience, New York, NY, USA, 1985.

[3] M. Lazard and P. Corvisier, "Inverse method for transient temperature estimation during machining," in *Proceedings of the 5th International Conference on Inverse Problems in Engineering: Theory and Practice*, Cambridge, UK, 2005.

[4] J. Yvonnet, D. Umbrello, F. Chinesta, and F. Micari, "A simple inverse procedure to determine heat flux on the tool in orthogonal cutting," *International Journal of Machine Tools and Manufacture*, vol. 46, no. 7-8, pp. 820–827, 2006.

[5] K. A. Woodbury, S. Duvvuri, Y. K. Chou, and J. Liu, "Use of evolutionary algorithms to determine tool heat fluxes in a machining operation," in *Proceedings of the Inverse Problems Design and Optimization Symposium (IPDO ')*, Miami Beach, Fla, USA, 2007.

[6] D. Ulutan, I. Lazoglu, and C. Dinc, "Three-dimensional temperature predictions in machining processes using finite difference method," *Journal of Materials Processing Technology*, vol. 209, no. 2, pp. 1111–1121, 2009.

[7] F. Samadi, F. Kowsary, and A. Sarchami, "Estimation of heat flux imposed on the rake face of a cutting tool: a nonlinear, complex geometry inverse heat conduction case study," *International Communications in Heat and Mass Transfer*, vol. 39, no. 2, pp. 298–303, 2012.

[8] N. Evangelista Luiz, *Usinabilidade do aço de corte fácil baixo carbono ao chumbo abnt 12l14 com diferentes níveis de elementos químicos residuais (cromo, níquel e cobre) [doctorate thesis]*, School of Mechanical Engineering, Federal University of Uberlândia, Uberlândia, Brazil, 2007.

[9] S. R. Carvalho, S. M. M. Lima e Silva, A. R. Machado, and G. Guimarães, "Temperature determination at the chip-tool interface using an inverse thermal model considering the tool and tool holder," *Journal of Materials Processing Technology*, vol. 179, no. 1–3, pp. 97–104, 2006.

[10] R. F. Brito, S. R. Carvalho, S. M. M. Lima e Silva, and J. R. Ferreira, "Thermal analysis in TIN and AL_2O_3 coated iso K10 cemented carbide cutting tools using Design of Experiment (DoE) methodology," *Journal of Machining and Forming Technologies*, vol. 3, pp. 1–12, 2011.

[11] R. F. Brito, S. R. D. Carvalho, S. M. M. D. Lima e Silva, and J. R. Ferreira, "Thermal analysis in coated cutting tools," *International Communications in Heat and Mass Transfer*, vol. 36, no. 4, pp. 314–321, 2009.

[12] P. F. B. Sousa, S. R. Carvalho, and G. Guimarães, "Dynamic observers based on Green's functions applied to 3D inverse thermal models," *Inverse Problems in Science and Engineering*, vol. 16, no. 6, pp. 743–761, 2008.

[13] Taylor Specials Steels Ltda, 2005, http://www.taylorspecial-steels.co.uk/pdfdownload/m35.pdf#search='M35%20thermal%20properties.

[14] P. A. Dearnley, "New technique for determining temperature distribution in cemented carbide cutting tools," *Metals Technology*, vol. 10, no. 6, pp. 205–214, 1983.

[15] E. M. Trent, *Metal Cutting*, Butterworths, London, UK, 2nd edition, 1984.

[16] A. R. Machado and M. B. Silva, *Usinagem dos Metais*, 8th edition, 2004.