Journal of Marine Science and Technology

Volume 24 | Issue 4 Article 3

APPLICATION OF A RESIDUAL-NORM-BASED ALGORITHM TO SOLVE ELLIPTIC BOUNDARY VALUE PROBLEMS

Yung-Wei Chen
Department of Marine Engineering, National Taiwan Ocean University, Keelung, Taiwan, R.O.C,
cyw0710@mail.ntou.edu.tw

Chun-Ming Chang
Department of Harbor and River Engineering, National Taiwan Ocean University, Keelung, Taiwan, R.O.C

Hui-Ming Fang
Department of Systems Engineering and Naval Architecture, National Taiwan Ocean University, Keelung, Taiwan, R.O.C.

Pai-Chen Guan
Department of Systems Engineering and Naval Architecture, National Taiwan Ocean University, Keelung, Taiwan, R.O.C.

APPLICATION OF A RESIDUAL-NORM-BASED ALGORITHM TO SOLVE ELLIPTIC BOUNDARY VALUE PROBLEMS

Acknowledgements

The author acknowledge the financial support of the National Science Council under contract number: NSC102 -2218-E-019-001.

Journal of Marine Science and Technology, Vol. 24, No. 4, pp. 690-701 (2016)
DOI: 10.6119/JMST-016-0115-1

APPLICATION OF A RESIDUAL-NORM-BASED ALGORITHM TO SOLVE ELLIPTIC BOUNDARY VALUE PROBLEMS

Yung-Wei Chen[1], Chun-Ming Chang[2], Hui-Ming Fang[2], and Pai-Chen Guan[3]

Key words: fictitious time integration (FTIM), boundary value problem, finite difference method (FDM).

ABSTRACT

In this study, we use a residual-norm-based algorithm (RNBA) to solve nonlinear elliptic boundary value problems (BVPs) on an arbitrary planar domain. For complex geometries, BVPs are very difficult and time-consuming to solve using conventional finite difference methods (FDMs). To overcome these problems, we apply a novel finite difference method (NFDM). By adding a fictitious rectangular domain and a bilinear function, we can easily treat geometrically complex boundary conditions. Then, through the use of the internal residual and the boundary residual, we can easily obtain the solution without the necessity of computing a matrix inverse. The RNBA avoids the oscillations that can occur in the manifold-based exponentially convergent algorithm (MBECA) by maintaining the manifold properties while guaranteeing convergence order greater than one. The accuracy and the convergence behaviour of this new method are demonstrated with several examples.

I. INTRODUCTION

Because of the rapid development of computer science and technology since 1980, numerical methods have become the method of choice for solving partial differential equations (PDEs). Currently, the most common numerical methods are the finite difference method (FDM), the finite element method (FEM), the boundary element method (BEM), and the meshless, or mesh-free, method. The FDM was the earliest method developed, and it can be easily combined with discretization techniques to solve engineering problems. However, if a quasi-linear PDE encounters a nonlinear problem in an arbitrary domain, both the geometric complexity and the nonlinearity encroach, and typically the conventional FDM requires special modifications to find the solution of such problems.

A variety of algorithms have been presented to address quasi-linear BVPs; examples include the mountain iteration algorithm, the scaling iterative algorithm, the monotone iterative algorithm, and the direct iterative algorithm, which were discussed by Chen et al. (2000). In general, a sequence of iterations is generated by various methods, and the sequence is not usually guaranteed to converge to the true solution. Many studies have presented numerical solutions of linear partial differential equations for BVPs. For example, the mesh-free local Galerkin method was developed and applied in computer modeling and simulation (Atluri et al., 1998ab; Castillo et al., 2000; Peraire et al., 2008). In addition, a mesh-free partition of unity method was proposed to deal with diffusion equations on complex domains (Eigel et al., 2010). More importantly, radial basis collocation method (Hu et al., 2005; Hu et al., 2008), Trefftz methods (Liu, 2007ab; Fan et al., 2011; Fan et al., 2012), and method of fundamental solutions (Jin, 2004; Wei et al., 2007) were recently applied to fix the problems of elliptic equations successfully. The numerical methods mentioned here are effective for linear problems. For nonlinear problems, however, they are less effective, and the number of iterations required is prohibitive.

The development of efficient algorithms for solving systems of nonlinear algebraic equations is a very important problem in the field of numerical methods. Recently, Liu (2008) proposed a novel method named the fictitious time integration method (FTIM). The FTIM solves a system of n nonlinear algebraic equations by introducing a fictitious time variable to form an augmented system in $(n + 1)$-dimensional space that is mathematically equivalent to the original system in n-dimensional space. The roots of the original algebraic equations are obtained by numerically integrating the resultant system of ordinary differential equations, and the inverses of the algebraic equations are not required. Furthermore, Liu (2009) demonstrated the use of the FTIM for solving m-point BVPs and two-dimensional, quasi-linear, elliptic BVPs. Several nu-

Paper submitted 03/07/15; revised 11/12/15; accepted 01/15/16. Author for correspondence: Yung-Wei Chen (e-mail: cyw0710@mail.ntou.edu.tw).
[1] Department of Marine Engineering, National Taiwan Ocean University, Keelung, Taiwan, R.O.C.
[2] Department of Harbor and River Engineering, National Taiwan Ocean University, Keelung, Taiwan, R.O.C.
[3] Department of Systems Engineering and Naval Architecture, National Taiwan Ocean University, Keelung, Taiwan, R.O.C.

merical examples including Laplace's equation, Poisson's equation, reaction diffusion, Helmholtz's equation, the minimal surface problem, and explosion equations were solved. It was shown that the FTIM can easily address nonlinear boundary value problems and provide highly accurate solutions.

Three methods have been developed that construct a spacetime manifold that transforms a vector function into a time-dependent scalar function while retaining the properties of a Newton-like method, the FTIM with a time-like variable (Liu et al., 2009), a scalar homotopy method (Ku et al., 2010) and the manifold-based exponentially convergent algorithm (MBECA) (Liu et al., 2012). These approaches can solve large systems of nonlinear algebraic equations (NAEs) without computing the inverse of the Jacobian matrix. However, some problems exist in these algorithms. For example, the convergence of the scalar-based homotopy method is very slow because it is difficult to satisfy the convergence criterion. For the MBECA and the FTIM with a time-like variable, the parameters representing viscous damping and the time-like function must be determined. When these parameters are larger than a certain fixed value, numerical instability will occur in the time integration. To avoid these problems, Chen et al. (2014) introduced a group-preserving scheme (GPS) for the fictitious time integration. Given the cone structure of the GPS and the MBECA and their manifold properties, the manifold path generated by the MBECA preserves the cone structure through the GPS with a weighting factor. More importantly, we can find that the weighting factor satisfies the Cauchy-Schwarz inequality. According to the plastic concept, the properties of the MBECA must satisfy the GPS cone condition. This means that the yielding surface under plastic theory must be kept on the surface of the cone such that the plastic flow is constrained according to the time integration path of the cone.

In order to overcome the numerical instability in the time integration, Liu and Atluri (2011) proposed the novel residual-norm-based algorithm (RNBA). The RNBA derives a gradient-flow system of nonlinear ordinary differential equations (ODEs) governing the evolution of space parameters with an independent, fictitious, time-like variable, where the residual error automatically decreases to zero. The RNBAs are purely iterative in nature and have the advantages of convergence fast, without involving the inversion of Jacobian matrix and suitable solving a large system of NAEs. According to the formula of RNBA, some optimal numerical methods, optimally generalization regularization method (OGRM), optimal multivector iterative algorithm (OMVIA) and doubly optimized solution (DOS), are proposed by Liu (2012, 2013, 2014) and applied to deal with large linear inverse problems with large noisy level. In this study, we will propose the RNBA combined with a novel finite difference method (NFDM) by Fan (2010). We apply this method to calculate the solutions of elliptic-type BVPs defined in arbitrary plane domains. By introducing fictitious time coordinates, the FDM associated with shape function with geometric complexity need not be treated in the same spatial domain but can be conveniently used to deal with any complicated shape within the problem domain. Then, modifying a weighting factor of GPS can increase computational efficiency and preserve the manifold of the system as the solution evolves without introducing a fictitious time variable.

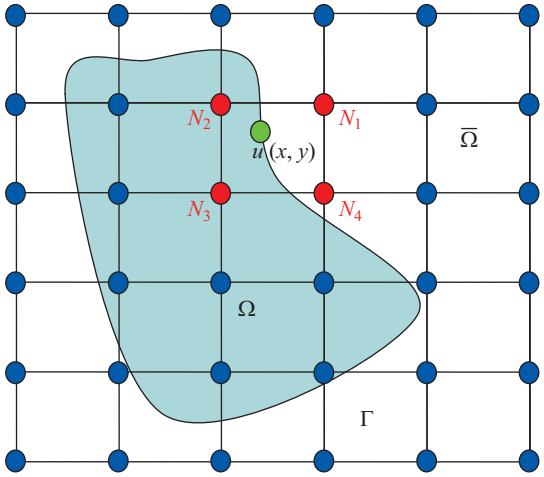

Fig. 1. Illustration of the computational domain.

The outline of this paper is as follows. Section 2 describes the BVPs and introduces the novel finite difference method (NFDM) and the residual-norm-based algorithm (RNBA). In Section 3, we demonstrate the approach on four numerical examples of nonlinear problems, and we compare the results with those of the original MBECA. In Section 4, we summarize the study and present our conclusions.

II. PROBLEM FORMULATION

1. The NFDM

In this study, we consider elliptic BVPs and their solution using the FDM. The following quasi-linear elliptic equation is considered:

$$\Delta u(x, y) = P(x, y, u_x, u_y, \cdots), (x, y) \in \Omega, \tag{1}$$

$$\Gamma_D u(x, y) + \Gamma_N u_n(x, y) = G(x, y), (x, y) \in \Gamma, \tag{2}$$

where Γ_D and Γ_N are coefficients, Δ is the Laplacian operator, Γ is the boundary of the problem domain Ω, and P and G are given functions. Here, the boundary Γ in polar coordinates can be described by a radius function, i.e., $\Gamma = \{(r,\theta) \mid r = \rho(\theta), \theta \in [0, 2\pi]\}$. To conveniently address an arbitrary computational domain using the FDM, we introduce an equidistant rectangle to enclose the problem domain Ω. Fig. 1 illustrates the arbitrary problem domain. Suppose that the rectangle is given by $\bar{\Omega} := [-a_x, a_x] \times [-b_y, b_y]$ such that

a_x and b_x cover the maximum dimensions of the problem domain in the x-axis and the y-axis, respectively. Then, we divide the rectangle $\bar{\Omega}$ into a uniform grid, with $\Delta x = 2a_x/(m-1)$ and $\Delta y = 2b_y/(m-1)$ in the x-direction and the y-direction, respectively. Furthermore, we let $u_{i,j}(t) = u(x_i, y_j, t)$ be the value of u at the grid point $(x_i, y_j) \in \bar{\Omega}$ evaluated at time t, where $x_i = (i-1)\Delta x$ and $y_i = (i-1)\Delta y$.

We approximate the Laplacian operator with finite differences and discretize the domain with a uniform grid. Then, Eq. (1) can be rewritten as follows:

$$\frac{u_{i+1,j} - 2u_{i,j} + u_{i-1,j}}{(\Delta x)^2} + \frac{u_{i,j+1} - 2u_{i,j} + u_{i,j-1}}{(\Delta y)^2} = P(x_i, y_j, \cdots),$$
$$i, j = 2, 3, 4, \ldots, m-1. \tag{3}$$

Although a uniformly spaced finite difference scheme can easily handle the governing equation, it is difficult to match the boundary condition in Eq. (2) exactly. Usually, an adaptive grid method is used to place the grid points on the perimeter of $\bar{\Omega}$ at the boundary. However, this may complicate the computations on an arbitrary domain because the boundary conditions are given on the boundary Γ of the domain Ω, not on the boundary of the rectangle $\bar{\Omega}$. Therefore, we must derive the governing equation of u_{ij} at the nodal points on the rectangular boundary.

To overcome this problem, we introduce the concept of a shape function to more easily address the boundary conditions. Let ξ and η be the local coordinates in the x-direction and the y-direction, respectively. We define ξ and η as:

$$\xi = 2\left(\frac{x - x_{i,j}}{\Delta x}\right) - 1, i, \quad j = 1, 2, 3, 4, \ldots, m, \tag{4}$$

$$\eta = 2\left(\frac{y - y_{i,j}}{\Delta y}\right) - 1, \quad i, j = 1, 2, 3, 4, \ldots, m, \tag{5}$$

where (x, y) are the global coordinates on the boundary and (x_{ij}, y_{ij}) denotes the coordinates of a grid point in $\bar{\Omega}$. Substituting Eqs. (4) and (5) into the shape function, the weighting coefficients for the boundary conditions can be expressed as follows:

$$\begin{aligned} N_1 &= \frac{1}{4}(1+\xi)(1+\eta) \\ N_2 &= \frac{1}{4}(1-\xi)(1+\eta) \\ N_3 &= \frac{1}{4}(1-\xi)(1-\eta) \\ N_4 &= \frac{1}{4}(1+\xi)(1-\eta) \end{aligned}, \tag{6}$$

where N_1, N_2, N_3, and N_4 denote the bilinear FEM shape functions. From Eqs. (2) and (6), the following equations are obtained to enforce the boundary condition when an element contains a boundary node:

$$\begin{aligned} \sum_{i=1}^{4} N_i^{b_1} u_{b_1} - G_{b_1} &= 0 \\ \sum_{i=1}^{4} N_i^{b_2} u_{b_2} - G_{b_2} &= 0 \\ &\vdots \\ \sum_{i=1}^{4} N_i^{b_n} u_{b_n} - G_{b_n} &= 0 \end{aligned}, \tag{7}$$

where $b_1, b_2, \ldots, b_n$ are the boundary points, $N^{b_1}, N^{b_2}, \ldots, N^{b_n}$ denote the bilinear FEM shape functions associated with the boundary points, and $G_{b_1}, G_{b_2}, \ldots, G_{b_n}$ denote the boundary conditions. Finally, Eqs. (3) and (7) can be rewritten as an algebraic equation as follows:

$$\mathbf{F}(\mathbf{x}) = 0, \tag{8}$$

where $\mathbf{x}$ is a vector of the unknown coefficients. When the residual errors of Eqs. (3) and (7) are both equal to zero, the solution of Eq. (8) is obtained. The details will be described in the following sections.

2. The RNBA

1) Construction of the Gradient Flow

We define a scalar function h that depends on the norm of the residual error in the solution to Eq. (8) and a monotonically increasing function $Q(t)$, where t is a pseudo-time:

$$h(\mathbf{x}, t) := \frac{1}{2} Q(t) \left\| \mathbf{F}(\mathbf{x}) \right\|^2, \tag{9}$$

and define a surface

$$h(\mathbf{x}, t) - C = 0. \tag{10}$$

Eq. (10) defines an invariant manifold in the space of $(\mathbf{x}, t)$. It is not necessary to specify the function $Q(t)$ a priori, where $\sqrt{2C/Q(t)}$ is as a measure of the residual error in Eq. (8) as a function of time. We require that $Q(t) > 0$ increases with t, and C is determined by the initial condition $\mathbf{x}(0) = \mathbf{x}_0$ as

$$C = \frac{1}{2} \left\| \mathbf{F}(\mathbf{x}) \right\|^2. \tag{11}$$

From the consistency condition, by taking the derivative of Eq. (10) with respect to t with $\mathbf{x} = \mathbf{x}(t)$, we have

$$\dot{h} = \frac{1}{2}\dot{Q}(t)\left\|\mathbf{F}(\mathbf{x})\right\|^2 + Q(t)(\mathbf{B}^{\mathrm{T}}\mathbf{F})\cdot\dot{\mathbf{x}} = 0\,, \tag{12}$$

where **B** is the Jacobian matrix, for which with the *ij*-entry is defined as $B_{ij} = \partial F_i \big/ \partial x_j$.

Assuming a "normality condition", the previous equation can expressed as follows:

$$\dot{\mathbf{x}} = -\lambda\frac{\partial h}{\partial \mathbf{x}} = -\lambda Q(t)\mathbf{B}^{\mathrm{T}}\mathbf{F}\,, \tag{13}$$

where λ is to be determined. Substituting Eq. (13) into (12), we obtain an expression for λ:

$$\lambda = \frac{\dot{Q}(t)\left\|\mathbf{F}\right\|^2}{2Q^2(t)\left\|\mathbf{B}^{\mathrm{T}}\mathbf{F}\right\|^2}\,. \tag{14}$$

Thus, we can obtain the gradient flow from Eqs. (13) and (14):

$$\dot{\mathbf{x}} = -q(t)\frac{\left\|\mathbf{F}\right\|^2}{\left\|\mathbf{B}^{\mathrm{T}}\mathbf{F}\right\|^2}\mathbf{B}^{\mathrm{T}}\mathbf{F}\,, \tag{15}$$

where

$$q(t) := \frac{\dot{Q}(t)}{2Q(t)}\,. \tag{16}$$

Hence, if $Q(t)$ is an increasing function of t, we have guaranteed convergence for the solution of the NAEs in Eq. (8) as $t\to\infty$ because

$$\left\|\mathbf{F}(\mathbf{x})\right\|^2 = \frac{2C}{Q(t)}\,. \tag{17}$$

2) Constraining the Solution to the Manifold

To ensure that the solution remains on the manifold defined by Eq. (17), we consider the evolution of **F** along the path **x**(t):

$$\dot{\mathbf{F}} = \mathbf{B}\dot{\mathbf{x}} = -q(t)\frac{\left\|\mathbf{F}\right\|^2}{\left\|\mathbf{B}^{\mathrm{T}}\mathbf{F}\right\|^2}\mathbf{A}\mathbf{F}\,, \tag{18}$$

where

$$\mathbf{A} := \mathbf{B}\mathbf{B}^{\mathrm{T}}\,. \tag{19}$$

Suppose that we use the Euler method to integrate Eq. (18):

$$\mathbf{F}(t+\Delta t) = \mathbf{F}(t) - \Delta t q(t)\frac{\left\|\mathbf{F}\right\|^2}{\left\|\mathbf{B}^{\mathrm{T}}\mathbf{F}\right\|^2}\mathbf{A}\mathbf{F}\,. \tag{20}$$

$$\mathbf{x}(t+\Delta t) = \mathbf{x}(t) - \Delta t q(t)\frac{\left\|\mathbf{F}\right\|^2}{\left\|\mathbf{B}^{\mathrm{T}}\mathbf{F}\right\|^2}\mathbf{B}^{\mathrm{T}}\mathbf{F}\,. \tag{21}$$

Taking the square-norms of both sides and using Eq. (16), we can obtain

$$\begin{aligned}\frac{2C}{Q(t+\Delta t)} = \frac{2C}{Q(t)} - 2\big(q(t)\Delta t\big)\frac{2C}{Q(t)}\frac{\mathbf{F}\cdot(\mathbf{A}\mathbf{F})}{\left\|\mathbf{B}^{\mathrm{T}}\mathbf{F}\right\|^2} \\ + \big(q(t)\Delta t\big)^2\frac{2C}{Q(t)}\frac{\left\|\mathbf{F}\right\|^2}{\left\|\mathbf{B}^{\mathrm{T}}\mathbf{F}\right\|^4}\mathbf{A}\mathbf{F}\end{aligned}\,. \tag{22}$$

Thus, the following scalar equation is obtained:

$$a(\Delta t)^2 - b\Delta t + 1 - \frac{Q(t)}{Q(t+\Delta t)} = 0\,, \tag{23}$$

where

$$a := q^2(t)\frac{\left\|\mathbf{F}\right\|^2\left\|\mathbf{A}\mathbf{F}\right\|^2}{\left\|\mathbf{B}^{\mathrm{T}}\mathbf{F}\right\|^4} \ge 1\,, \tag{24}$$

$$b := 2q(t)\,. \tag{25}$$

Substituting Eqs. (24) and (25) into Eq. (23), we have

$$a(\Delta t)^2 - b\Delta t + 1 - \frac{Q(t)}{Q(t+\Delta t)} = 0\,, \tag{26}$$

where $S = Q(t)/Q(t+\Delta t)$ and

$$A_0 := \frac{\left\|\mathbf{F}\right\|^2\left\|\mathbf{A}\mathbf{F}\right\|^2}{\left\|\mathbf{B}^{T}\mathbf{F}\right\|^4} \ge 1\,, \tag{27}$$

From Eq. (27) and the Cauchy-Schwarz inequality, we can write

$$\left\|\mathbf{B}^{T}\mathbf{F}\right\|^2 = \mathbf{F}\cdot\mathbf{A}\mathbf{F} \le \left\|\mathbf{F}\right\|\left\|\mathbf{A}\mathbf{F}\right\|\,. \tag{28}$$

When using the MBECA with $Q(t)$ fixed, oscillations may occur. Consequently, we let $Q(t)$ be automatically determined by the algorithm. From Eq. (26), we let

$$S = A_0(q\Delta t)^2 - 2(q\Delta t) + 1 = \frac{Q(t)}{Q(t+\Delta t)}. \tag{29}$$

To obtain the minimum of S, we take the derivative of Eq. (29) with respect to Δt .Setting the derivative equal to zero and solving for Δt gives

$$\Delta t = \frac{1}{qA_0}. \tag{30}$$

Substituting Eq. (30) into Eq. (29), we obtain the minimum value of S:

$$S = 1 - \frac{1}{A_0}. \tag{31}$$

From Eqs. (17) and (29), we can write the following expression:

$$\frac{\left\|\mathbf{F}(t+\Delta t)\right\|}{\left\|\mathbf{F}(t)\right\|} = \sqrt{S}. \tag{32}$$

From Eqs. (32) and (31), it follows that the ratio of two consecutive residual errors is less than one.

3) The RNBA

Substituting the value of Δt from Eq. (30) into Eq. (21), we obtain the expression

$$\begin{aligned}\mathbf{x}(t+\Delta t) &= \mathbf{x}(t) - \frac{1}{A_0}\frac{\left\|\mathbf{F}\right\|^2}{\left\|\mathbf{B}^{\mathrm{T}}\mathbf{F}\right\|^2}\mathbf{B}^{\mathrm{T}}\mathbf{F} \\ &= \mathbf{x}(t) - \frac{\left\|\mathbf{B}^{\mathrm{T}}\mathbf{F}\right\|^2}{\left\|\mathbf{A}^{\mathrm{T}}\mathbf{F}\right\|^2}\mathbf{B}^{\mathrm{T}}\mathbf{F}\end{aligned}, \tag{33}$$

where $\mathbf{B}^{\mathrm{T}}\mathbf{F}$ represents the gradient vector and $\left\|\mathbf{B}^{\mathrm{T}}\mathbf{F}\right\|^2 \mathbf{B}^{\mathrm{T}}\mathbf{F} \Big/ \left\|\mathbf{A}^{\mathrm{T}}\mathbf{F}\right\|^2$ is a regularized gradient vector. It can be observed that the preceding algorithm, no special parameters or a time step (Δt) are required. To improve the convergence, a weighting factor ω is included in Eq. (33). From Eq. (26), we can solve for Δt:

$$\Delta t = \frac{1+\sqrt{1-(1-S)A_0}}{qA_0}, \text{ if } 1-(1-S)A_0 \ge 0. \tag{34}$$

$$\Delta t = \frac{1}{qA_0}, \text{ if } 1-(1-S)A_0 < 0. \tag{35}$$

Let

$$1-(1-S)A_0 = \gamma^2 \ge 0, \; S = 1 - \frac{1-\gamma^2}{A_0}. \tag{36}$$

Thus, we have

$$\Delta t = \frac{1-\gamma}{qA_0}. \tag{37}$$

Substituting Eqs. (35) and (37) into Eq. (21), we obtain the following expression:

$$\mathbf{x}(t+\Delta t) = \mathbf{x}(t) - \omega\frac{\left\|\mathbf{B}^{\mathrm{T}}\mathbf{F}\right\|^2}{\left\|\mathbf{A}^{\mathrm{T}}\mathbf{F}\right\|^2}\mathbf{B}^{\mathrm{T}}\mathbf{F}, \tag{38}$$

where

$$\omega = 1-\gamma \quad \text{if } 1-(1-S)A_0 \ge 0. \tag{39}$$

$$\omega = 1 \quad \text{if } 1-(1-S)A_0 < 0. \tag{40}$$

Here, $0 \le \gamma < 1$ is a parameter and $0 \le \omega \le 1$ is weighting factor.

4) GPS for a System of Differential Equations

The GPS (Liu, 2001) maintains the properties of the space-time manifold and automatically changes the weighting factor. We can rewrite Eq. (33) in vector form:

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x},t) = \frac{\left\|\mathbf{B}^{\mathrm{T}}\mathbf{F}\right\|^2}{\left\|\mathbf{A}^{\mathrm{T}}\mathbf{F}\right\|^2}\mathbf{B}^{\mathrm{T}}\mathbf{F}, \; \mathbf{x} \in \mathrm{R}^n, \; t > 0, \tag{41}$$

where n is the number of algebraic equations.

The GPS can preserve the internal symmetry group of the system of ODEs. Eq. (41) can be augmented to form the following dynamic system of dimension n+1:

$$\frac{d}{dt}\begin{bmatrix}\mathbf{x} \\ \left\|\mathbf{x}\right\|\end{bmatrix} = \begin{bmatrix}\mathbf{0}_{n\times n} & \dfrac{\mathbf{F}(\mathbf{x},t)}{\left\|\mathbf{x}\right\|} \\ \dfrac{\mathbf{F}^{\mathrm{T}}(\mathbf{x},t)}{\left\|\mathbf{x}\right\|} & 0\end{bmatrix}\begin{bmatrix}\mathbf{x} \\ \left\|\mathbf{x}\right\|\end{bmatrix}. \tag{42}$$

The first n rows in Eq. (42) are from Eq. (41), and the last row in Eq. (42) provides a Minkowskian structure for the augmented state vector. Eq. (42) can be rewritten as:

$$\dot{\mathbf{X}} = \mathbf{A}\mathbf{X}. \tag{43}$$

With some efforts, the GPS can be shown as follows:

$$\mathbf{x}_{k+1} = \mathbf{x}_k + \omega_k \mathbf{F}_k , \tag{44}$$

$$\left\|\mathbf{x}_{k+1}\right\| = a_k \left\|\mathbf{x}_k\right\| + \frac{b_k}{\left\|\mathbf{F}_k\right\|} \mathbf{F}_k \cdot \mathbf{x}_k , \tag{45}$$

where

$$\omega_k := \frac{b_k \left\|\mathbf{x}_k\right\| \left\|\mathbf{F}_k\right\| + (a_k - 1)\mathbf{F}_k \cdot \mathbf{x}_k}{\left\|\mathbf{F}_k\right\|} . \tag{46}$$

The group properties are preserved in this scheme for all $\Delta t > 0$, and thus it is called a group-preserving scheme. However, using Eq. (46) in Eq. (38) can cause instability because Eq. (46) cannot easily satisfy the cone and Lie group properties. We can modify Eq. (44) to create a residual-norm algorithm with an adaptive step size as follows:

$$\omega = 1 - \gamma \quad \text{if} \quad 1 - (1 - S) A_0 \geq 0 . \tag{47}$$

The parameter $0 \leq \alpha < 1$ controls the residual error in the solution, and β is a fictitious time step size. At initialization, $\left\|\mathbf{x}\right\|$ is equal to zero and δ is a constant.

III. NUMERICAL EXAMPLES

In this section, we solve four examples to test the convergence and the stability of the proposed numerical method. In the following examples, m is the number of grid points in the x- and y-directions, n is the total number of boundary points, $\mathbf{x}_0$ is the unknown vector, and ε is the convergence criterion. Algorithm 1 uses the weighting factor given by the GPS, algorithm 2 uses the weighting factor $\omega = 1 - \gamma$, and algorithm 3 uses the weighting factor $\omega = 1$.

Example 1

In this example, we consider a two-dimensional arbitrary domain as shown in Fig. 2 under a Poisson's equation, expressed as follows:

$$\Delta u + 2u = 2e^x \cos y + 20, \quad (x, y) \in \Omega . \tag{48}$$

The boundary conditions are given as follows:

$$u(x, y) = e^x \cos y + 10, \quad (x, y) \in \Gamma, \qquad 0 \leq \theta \leq \pi . \tag{49}$$

The analytical solution for this problem is:

$$u(x, y) = e^x \cos y + 10, \quad (x, y) \in \Gamma \bigcup \Omega. \tag{50}$$

We used the RNBA to solve this problem, where the values

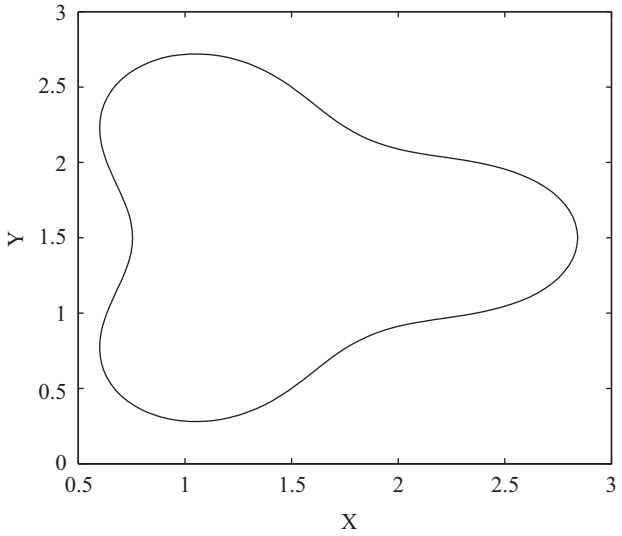

Fig. 2. The domain for example 1.

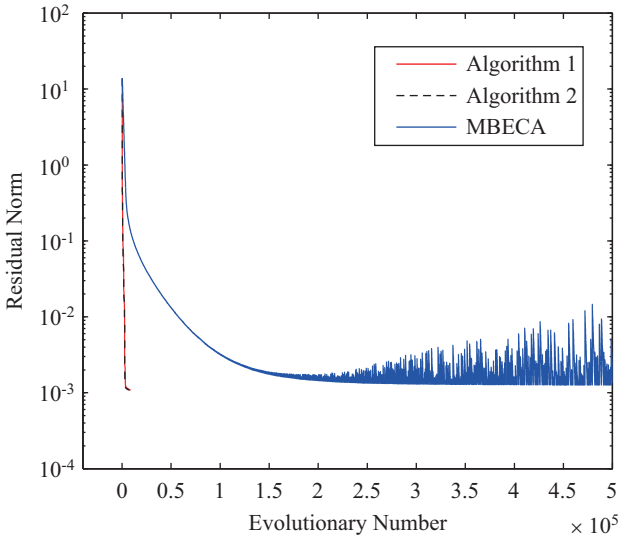

Fig. 3. Comparison of the residual norms of three algorithms for example 1.

of the parameters were $m = 20$, $n = 160$, $\mathbf{x}_0 = 0.5$, $\varepsilon = 1.09 \times 10^{-3}$, $\gamma = 0.08$, $\alpha = 0.1$ and $\beta = 0.0001$. The residual norms versus the number of steps are shown in Fig. 3, where the red, black and blue lines denote algorithm 1, algorithm 2 and the MBECA, respectively. In Fig. 3, we can observe that the residual norms of algorithm 1 and algorithm 2 decrease rapidly, and the residual errors satisfy the convergence criterion at 5.937×10^3 steps for algorithm 1 and 7.216×10^3 steps for algorithm 2. The residual norm, A_0 and S for algorithms 1 and 2 are shown in Figs. 4(a) and (b), respectively. The analytic solution is shown in Fig. 5(a), and the numerical errors are shown in Figs. 5(b) and (c), respectively. Both algorithm 1 and algorithm 2 give rather accurate results, with maximum errors less than 3.87×10^{-2}.

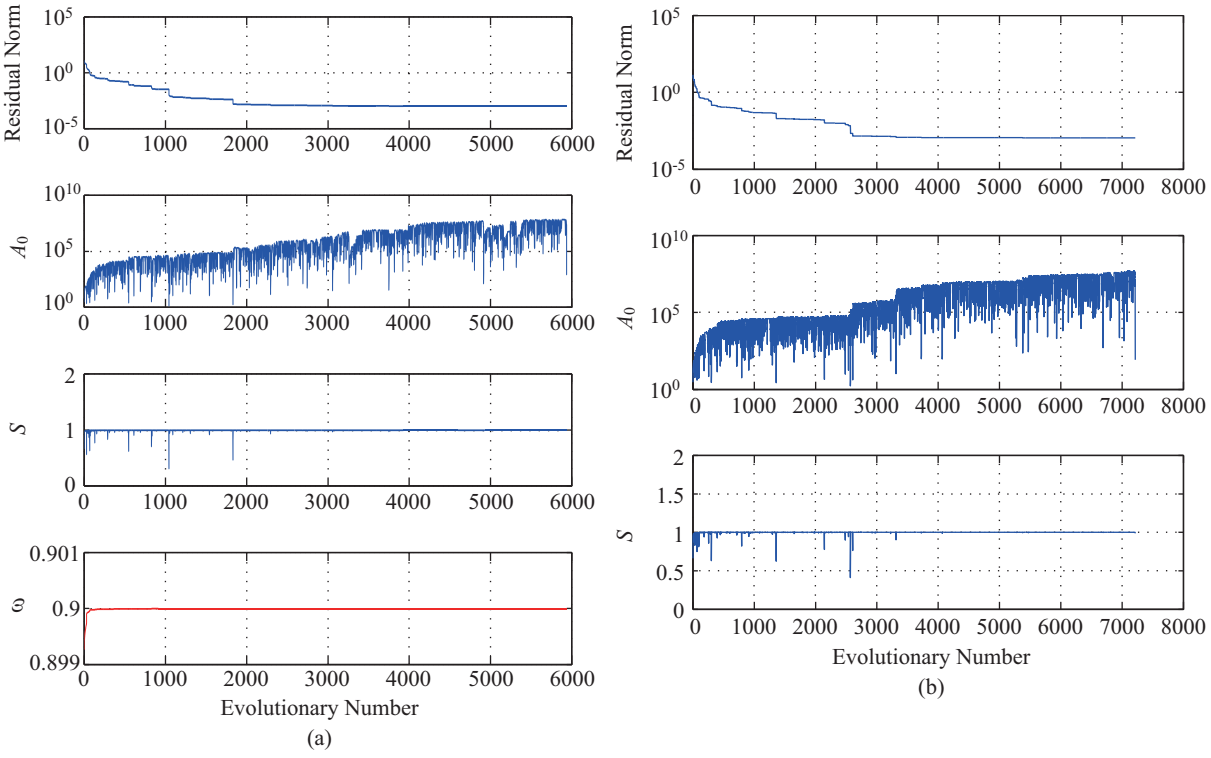

Fig. 4. Residual norm, A_0, S and ω for example 1: (a) algorithm 1, and (b) algorithm 2.

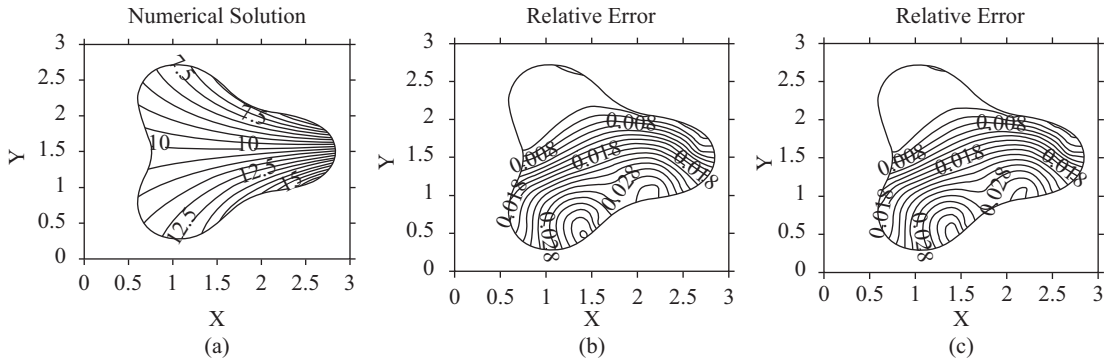

Fig. 5. Solutions to example 1, the potential for a linear equation with mixed boundary conditions: (a) analytic solution, (b) relative error of algorithm 1, and (c) relative error of algorithm 2.

Example 2

In this example, the domain of interest is circular, and it can be expressed as follows:

$$\Omega = \left\{ \begin{array}{l} (x,y) \mid x = 0.48\cos\theta + 0.5, \\ \qquad y = 0.48\sin\theta + 0.5, \quad 0 \le \theta \le 2\pi \end{array} \right\}. \tag{51}$$

We include the convection and diffusion terms in the governing equation, which is expressed as follows:

$$\nabla^2 u + xy\frac{\partial u}{\partial x} + x^2\frac{\partial u}{\partial y} = b, \qquad (x,y) \in \Omega, \tag{52}$$

where

$$b = \left(-\cos x - 18\cos 3y\right) + xy\left(-\sin+1\right) + x^2\left(-6\sin 3y + 1\right). \tag{53}$$

The boundary conditions are

$$u(x,y) = \cos x + 2\cos y + x + y + 3, \quad (x,y) \in \Gamma, \quad 0 \le \theta \le \frac{3\pi}{2}, \tag{54}$$

$$u_n(x,y) = \left(1-\sin x\right) nx - \left(1+\sin y\right) ny, \quad (x,y) \in \Gamma, \quad \frac{3\pi}{2} \le \theta \le 2\pi. \tag{55}$$

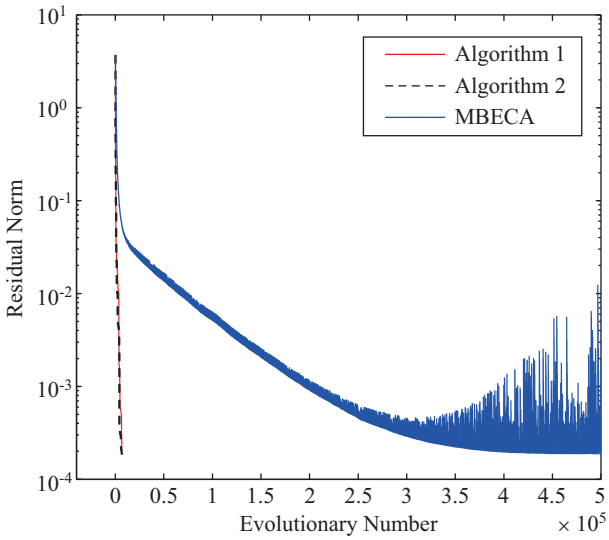

Fig. 6. Comparison of the residual norms of three algorithms for example 2.

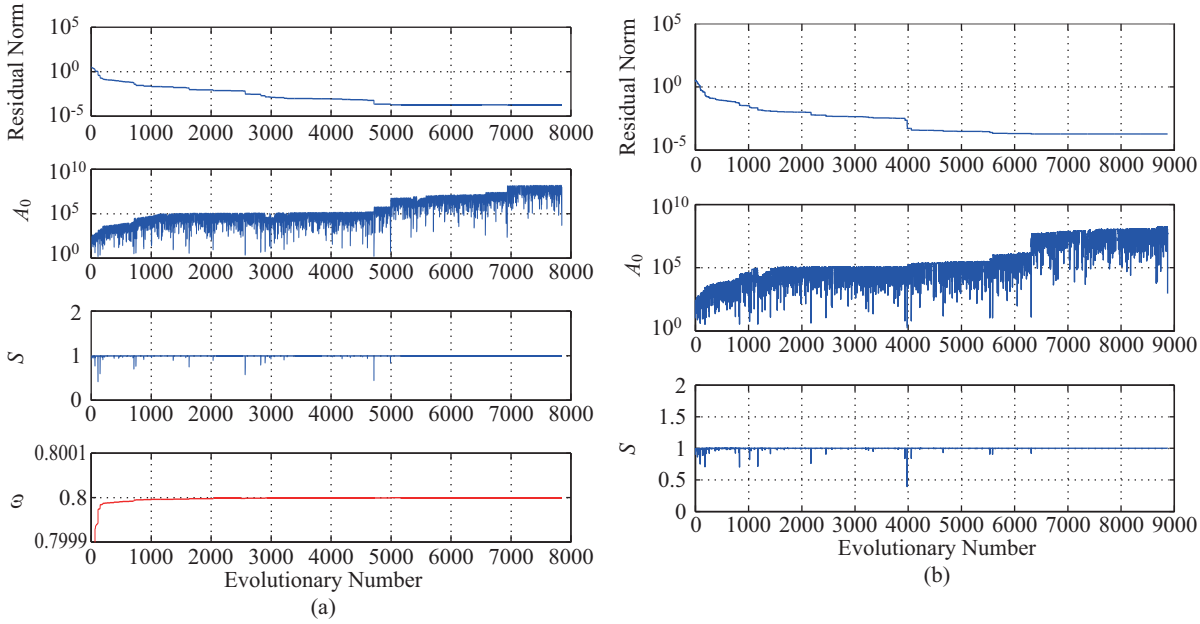

Fig. 7. Residual norm, A_0, S and ω for example 2: (a) algorithm 1, and (b) algorithm 2.

The values of the parameters in the algorithms are $m = 20$, $n = 160$, $\mathbf{x}_0 = 1.0$, $\varepsilon = 1.9\times10^{-4}$, $\gamma = 0.08$, $\alpha = 0.2$ and $\beta = 0.001$. The residual norms versus the number of steps are shown in Fig. 6, where the red, black and blue lines denote algorithm 1, algorithm 2 and the MBECA, respectively. The Figure shows that the residual norms decrease rapid with converging in 7.848×10^3 steps for algorithm 1 and 8.876×10^3 steps for algorithm 2. From the results, it can be observed that the convergence of the MBECA varies very dramatically as the number of steps increases. In contrast, the convergence rates of algorithms 1 and 2 are stable despite the addition of the convection and diffusion terms in the governing equation.

The residual norm, A_0 and S for algorithms 1 and 2 are shown in Figs. 7(a) and (b), respectively. In the Figures, it can be observed that A_0 is much greater than one when S is equal to one, and the residual norm clearly. The analytic solution is shown in Fig. 8(a), and the relative errors of the solutions from algorithms 1and 2 are plotted in Figs. 8(b) and (c), where it can be observed that the maximum error was less than 2.223×10^{-2}.

Example 3

In this example, we verify that the proposed scheme can accommodate a multiply connected domain with genus 1. The

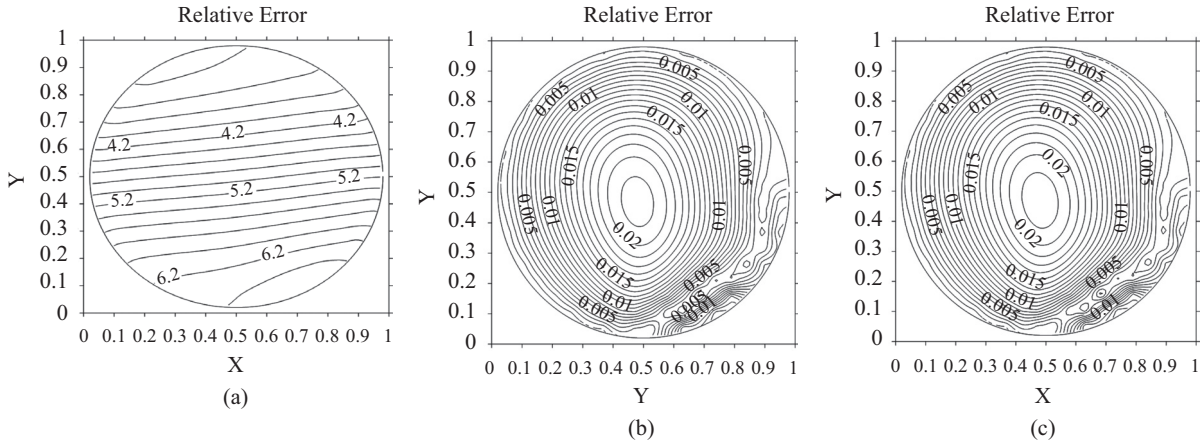

Fig. 8. Solutions to example 2, the potential for a linear equation with mixed boundary conditions: (a) analytic solution, (b) relative error of algorithm 1, and (c) relative error of algorithm 2.

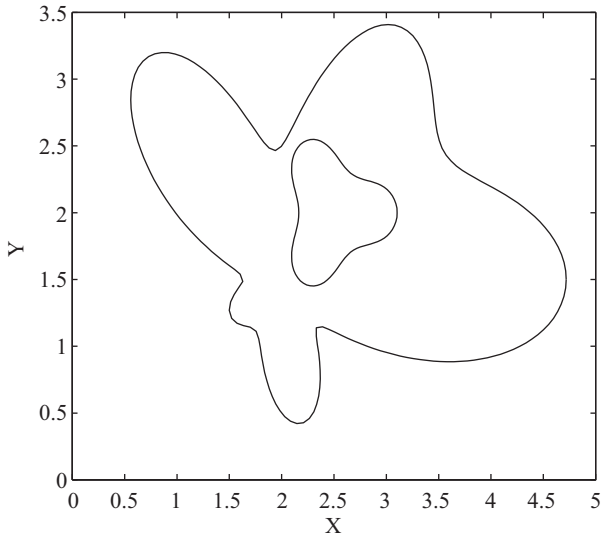

Fig. 9. The domain for example 3.

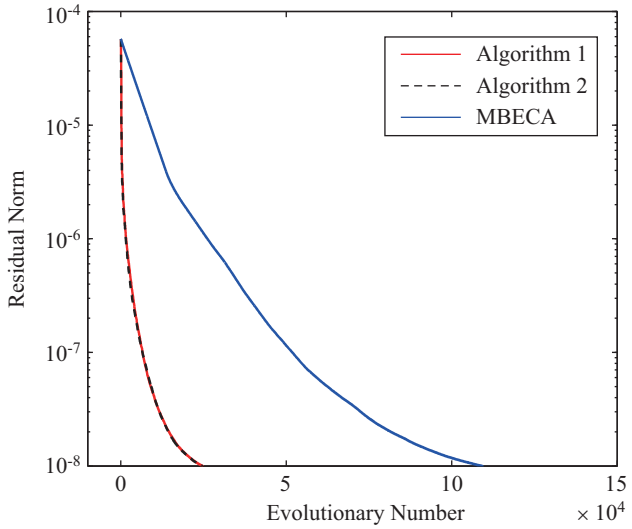

Fig. 10. Comparison of the residual norms of three algorithms for example 3.

irregularly shaped domain is plotted in Fig. 9, where the contour is given by

$$\Gamma_1 = \{(x,y) \mid x = \rho_1 \cos\theta + 2,\ \ y = \rho_1 \sin\theta + 1.5\},\quad (x,y) \in \Gamma,\ \ 0 \le \theta \le 2\pi, \tag{56}$$

$$\Gamma_2 = \{(x,y) \mid x = \rho_2 \cos\theta + 2.5,\ \ y = \rho_2 \sin\theta + 2\},\quad (x,y) \in \Gamma,\ \ 0 \le \theta \le 2\pi \tag{57}$$

with

$$\rho_1 = \exp(\sin(\theta))\sin^2(2\theta) + \exp(\cos(\theta))\cos^2(2\theta), \tag{58}$$

$$\rho_2 = 0.45 \times \left(\cos(3\theta) + \sqrt{2 - \sin^2(3\theta)}\right)^{\frac{1}{3}}. \tag{59}$$

We wish to solve the following nonlinear diffusion equation:

$$\nabla^2 u = 4u^3\left(x^2 + y^2 + a^2\right),\ \ a = 8,\ \ (x,y) \in \Omega. \tag{60}$$

The exact solution for this problem is:

$$u(x,y) = \frac{-1}{x^2 + y^2 - a^2},\ \ (x,y) \in \Gamma \cup \Omega, \tag{61}$$

where $\Gamma = \Gamma_1 + \Gamma_2$.

The values of the parameters were m = 16, n = 300, $\mathbf{x}_0 = 0.01$, $\varepsilon = 1 \times 10^{-8}$, $\gamma = 0.08$, $\alpha = 0.08$ and $\beta = 0.1$. The residual norms versus the number of steps for the three algorithms are shown in Fig. 10, where the red, black and blue lines denote the results from algorithm 1, algorithm 2 and the

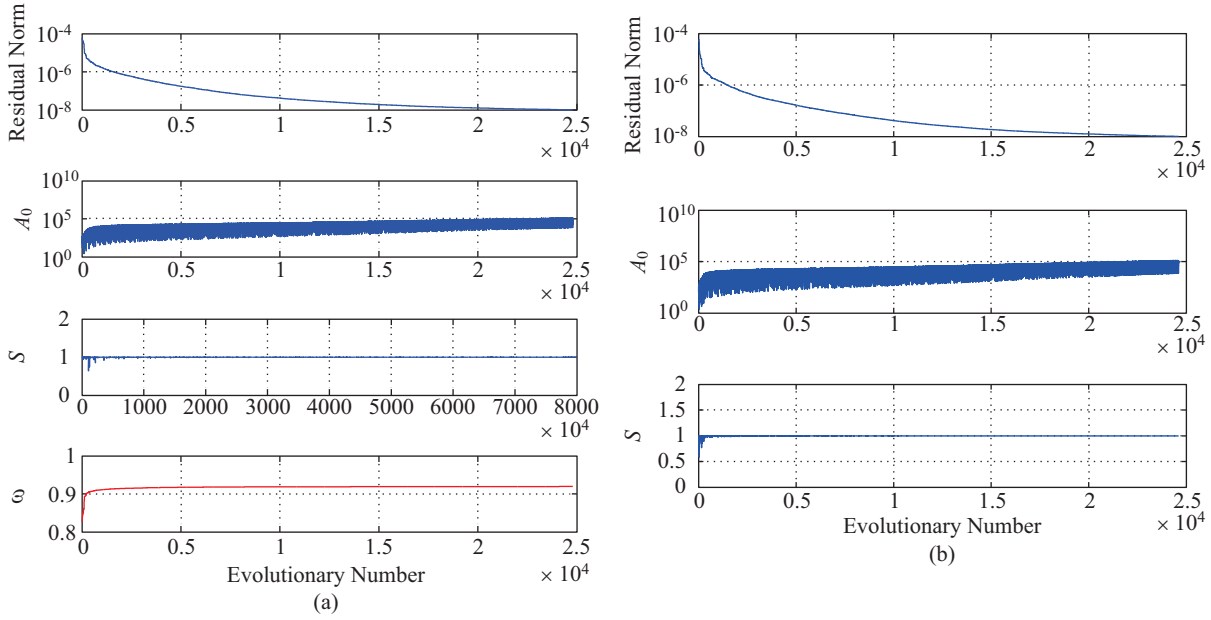

Fig. 11. sidual norm, A_0, S and ω for example 3: (a) algorithm 1, and (b) algorithm 2.

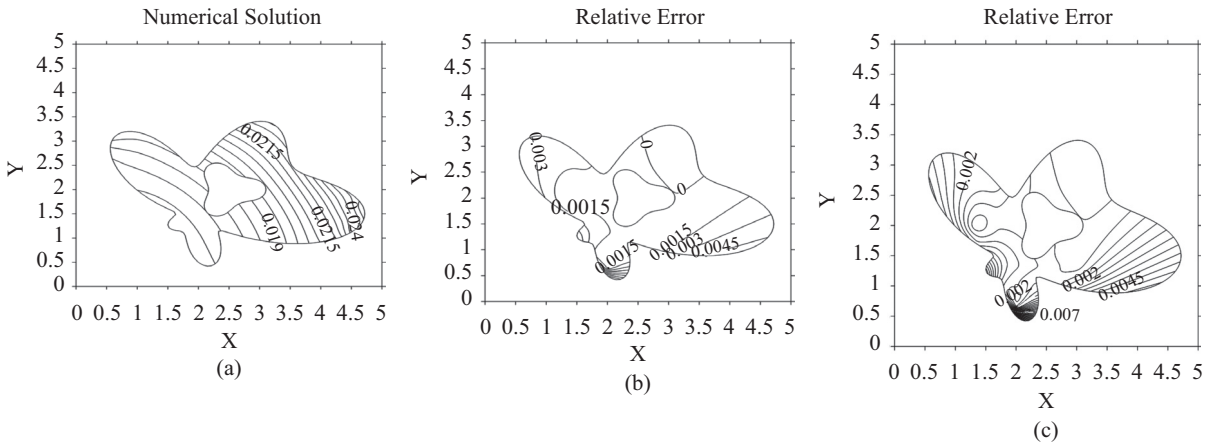

Fig. 12. Solutions to example 3, the potential for a nonlinear equation with Dirichlet-type boundary conditions: (a) analytic solution, (b) relative error of algorithm 1, and (c) relative error of algorithm 2.

MBECA, respectively. The Figure shows that the residual errors for algorithms 1 and 2 satisfy the convergence criterion at 2.4803×10^4 and 2.4609×10^4 steps, respectively. The residual norm, A_0 and S are shown for algorithms 1 and 2 in Figs. 11(a) and (b), respectively. In the Figures, it can be observe that A_0 is much greater than one when S is equal to one, and the residual norm is less than 1×10^{-8}.

The analytic solution is shown in Fig. 12(a), and the relative errors of the solutions from algorithms 1 and 2 are plotted in Figs. 12(b) and (c), respectively, where it can be observed that the maximum error was less than 3.914×10^{-2}.

Example 4

To test the efficiency of the proposed method in solving a nonlinear equation, the following nonlinear Poisson equation is considered:

$$\nabla^2 u + u^2 = b, \quad (x, y) \in \Omega, \tag{62}$$

where

$$b = (e^x \cos y + x + y + 1)^2 . \tag{63}$$

We assume the same geometry as in Example 1; the boundary conditions are

$$u(x, y) = e^x \cos y + x + y + 1, \quad 0 \le \theta \le 2\pi, \quad (x, y) \in \Gamma, \tag{64}$$

The analytical solution for this problem is

$$u(x, y) = e^x \cos y + x + y + 1, \quad (x, y) \in \Gamma \cup \Omega . \tag{65}$$

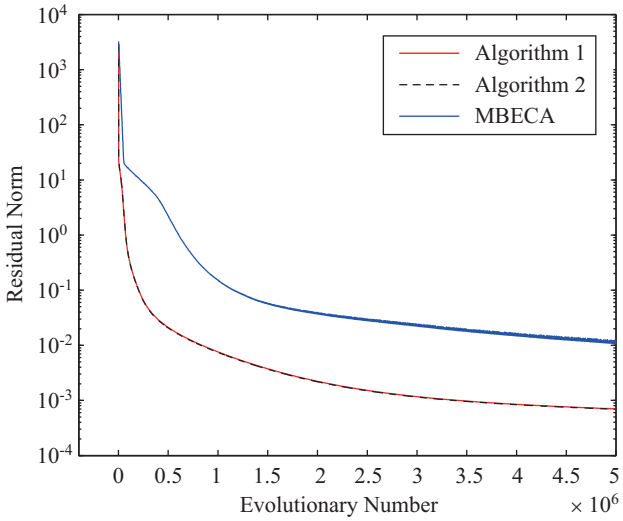

Fig. 13. Comparison of the residual norms of three algorithms: a comparison for example 4.

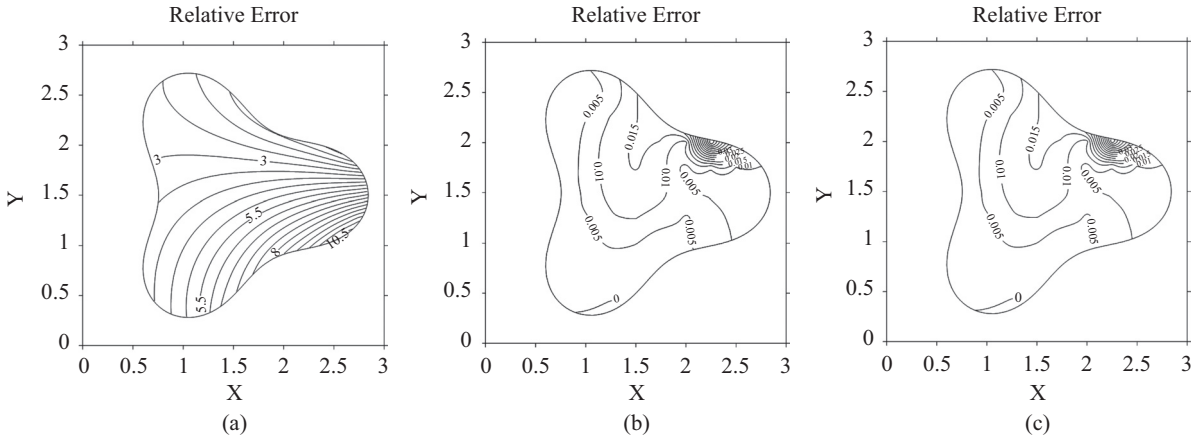

Fig. 14. Solutions to example 4, the potential for a nonlinear equation: (a) analytic solution, (b) relative error of algorithm 1, and (c) relative error of algorithm 2.

The values of the parameters were m = 13, n = 160, $\mathbf{x}_0 = 0.01$, $\varepsilon = 1\times10^{-6}$, $\gamma = 0.08$, $\alpha = 0.08$ and $\beta = 0.1$. The residual norms versus the number of steps for the three algorithms are shown in Fig. 13, where the red, black and blue lines denote the results from algorithm 1, algorithm 2 and the MBECA, respectively. It can be observed that the RNBA converges faster than the MBECA. The analytic solution of the equation is shown in Fig. 14(a), and the relative errors of the solutions from algorithms 1and 2 are plotted in Figs. 14(b) and (c), respectively, showing that the maximum error was less than 7.426×10^{-2}.

CONCLUSIONS

In this study, we developed the RNBA to solve nonlinear elliptic boundary value problems on arbitrary planar domains. Elliptic BVPs with nonlinear PDEs and geometrically complex domains cannot easily be solved using the FDM. To overcome these obstacles, we introduced the concepts of the internal residual and the boundary residual by imposing a fictitious rectangular domain and a bilinear function that we can easily manipulate to address any boundary condition. As a result, we are only required to numerically integrate the ODEs until the residuals are reduced to a specified value to obtain a solution. Additionally, the RNBA is simple to implement and does not require the inverse of the Jacobian matrix to be calculated. From the results of several numerical examples, we can conclude that this new approach can successfully solve problems with linear and nonlinear equations and converges much faster than the MBECA. In summary, this scheme has several advantages over the conventional one, including of high efficiency and stability.

ACKNOWLEDGMENTS

The author acknowledge the financial support of the Na-

tional Science Council under contract number: NSC102 -2218-E-019-001.

REFERENCES

Atluri, S. N. and T. Zhu (1998a). A new meshless local Petrov-Galerkin (MLPG) approach in computational mechanics. Computational Mechanics 22, 117-127.

Atluri, S. N. and T. Zhu (1998b). A new meshless local Petrov-Galerkin (MLPG) approach to nonlinear problems in computer modeling and simulation. CMES: Computer Modeling in Engineering and Sciences 3, 187-196.

Castillo, P., B. Cockburn, I. Perugia and D. Schötzau (2000). An a priori error analysis of the local discontinuous Galerkin method for elliptic problems. SIAM Journal on Numerical Analysis 38, 1676-1706.

Chen, G. and J. Zhou (2000). Algorithms and visualization for solutions of nonlinear elliptic equations. International Journal of Bifurcation and Chaos 10, 1565-1612.

Chen, Y. W., C. M. Chang, C. S. Liu and J. R. Chang (2014). Application of a manifold-based exponentially convergent algorithm to solve elliptic boundary-value problems. IMA Journal Numerical Analysis 34, 362-389.

Eigel, M., E. George and M. Kirkilionis (2010). A mesh-free partition of unity method for diffusion equations on complex domains. IMA Journal of Numerical Analysis 30, 629-653.

Fan, C. M., H. F. Chan, C. L. Kuo and W. C. Yeih (2012). Numerical solution of boundary detection problems using modified collocation Trefftz method and exponentially convergent scalar homotopy algorithm. Engineering Analysis with Boundary Elements 36, 2-8.

Fan, C. M., H. H. Li and C. L. Kuo (2011). The modified collocation Trefftz method and Laplacian decomposition for solving two-dimensional Stokes problems. Journal of Marine Science and Technology 19, 522-530.

Fan, W. C. (2010). Solving the elliptic type PDEs by a combination of a novel finite difference method with the manifold-based exponentially convergent algorithm. Master Thesis, National Taiwan Ocean University, Taiwan.

Hu, H. Y. and J. S. Chen (2008). Radial basis collocation method and quasi-Newton iteration for nonlinear elliptic problems. Numerical Methods for Partial Differential Equations 24, 991-1017.

Hu, H. Y., Z. C. Li and A. H. D. Cheng (2005). Radial basis collocation methods for elliptic boundary value problems. Computers & Mathematics with Applications 50, 289-320.

Jin, B. (2004). A meshless method for the Laplace and biharmonic equations subjected to noisy boundary data. CMES: Computer Modeling in Engineering and Sciences 6, 253-262.

Ku, C. Y., W. C. Yeih and C. S. Liu (2010). Solving non-linear algebraic equations by a scalar Newton-homotopy continuation method. International Journal Nonlinear Science and Numerical Simulation 11, 435-450.

Li, Z. C., T. T. Lu, H. Y. Hu and A. H. D. Cheng (2005). Trefftz and collocation methods, WIT Press Southampton, Boston.

Liu, C. S. (2001). Cone of non-linear dynamical system and group preserving schemes. International Journal of Non-Linear Mechanics 36, 1047-1068.

Liu, C. S. (2007a). A modified Trefftz method for two-dimensional Laplace equation considering the domain's characteristic length. CMES: Computer Modeling in Engineering and Sciences 21, 53-66.

Liu, C. S. (2007b). An effectively modified direct Trefftz method for 2D potential problems considering the domain's characteristic length. Engineering Analysis with Boundary Elements 31, 983-993.

Liu, C. S. (2008). A fictitious time integration method for two-dimensional quasilinear elliptic boundary value problems. CMES: Computer Modeling in Engineering and Sciences 33, 179-198.

Liu, C. S. (2009). A fictitious time integration method for solving *m*-point boundary value problems. CMES: Computer Modeling in Engineering and Sciences33, 125-154.

Liu, C. S. (2012). A manifold-based exponentially convergent algorithm for solving non-linear partial differential equations. Journal of Marine Science and Technology 20, 441-449.

Liu, C. S. (2012). Optimally generalized regularization methods for solving linear inverse problems. Computers Materials and Continua 29, 103-127.

Liu, C. S. (2013). An optimal multi-vector iterative algorithm in a Krylov subspace for solving the ill-posed linear inverse problems. Computers Materials and Continua 33, 175-198.

Liu, C. S. (2014). A doubly optimized solution of linear equations system expressed in an affine Krylov subspace. Journal of Computational and Applied Mathematics 260, 375-394.

Liu, C. S. and S. N. Atluri (2011). Simple "residual-norm" based algorithms, for the solution of a large system of non-linear algebraic equations, which converge faster than the newton's method. CMES: Computer Modeling in Engineering and Sciences 71, 279-304.

Liu, C. S., W. C. Yeih, C. L. Kuo and S. N. Atluri (2009). A scalar homotopy method for solving an over/under-determined system of non-linear algebraic equations. CMES: Computer Modeling in Engineering and Sciences 53, 47-71.

Peraire, J. and P. O. Persson (2008). The compact discontinuous Galerkin (CDG) method for elliptic problems. SIAM Journal on Numerical Analysis 30, 1806-1824.

Wei, T., Y. C. Hon and L. Ling (2007). Method of fundamental solution with regularization technique for Cauchy problems of elliptic operators. Engineering Analysis with Boundary Elements 31, 163-175.