Journal of Marine Science and Technology

Volume 22 | Issue 5 Article 7

A RESIDUAL-NORM BASED ALGORITHM FOR SOLVING DETERMINATE/ INDETERMINATE SYSTEMS OF NON-LINEAR ALGEBRAIC EQUATIONS

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

Recommended Citation

Chen, Yung-Wei (2014) "A RESIDUAL-NORM BASED ALGORITHM FOR SOLVING DETERMINATE/INDETERMINATE SYSTEMS OF NON-LINEAR ALGEBRAIC EQUATIONS," *Journal of Marine Science and Technology*: Vol. 22: Iss. 5, Article 7.
DOI: 10.6119/JMST-013-0912-1
Available at: https://jmstt.ntou.edu.tw/journal/vol22/iss5/7

A RESIDUAL-NORM BASED ALGORITHM FOR SOLVING DETERMINATE/ INDETERMINATE SYSTEMS OF NON-LINEAR ALGEBRAIC EQUATIONS

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. 22, No. 5, pp. 583-597 (2014)
DOI: 10.6119/JMST-013-0912-1

A RESIDUAL-NORM BASED ALGORITHM FOR SOLVING DETERMINATE/INDETERMINATE SYSTEMS OF NON-LINEAR ALGEBRAIC EQUATIONS

Yung-Wei Chen

Key words: fictitious time integration method (FTIM), residual-norm based algorithm (RNBA), Hopf bifurcation, Intermittency.

ABSTRACT

In this paper, a residual-norm based algorithm (RNBA) is applied to solve determinate/indeterminate systems of non-linear algebraic equations. The RNBA is derived from a manifold and defined in terms of a squared residual norm and a fictitious time variable, from which a robust iterative algorithm with either fixing or adjusting parameters can be obtained. Besides, some convergent indexes, such as manifold factor *A0* and ratio of residual errors *S*, are defined to indicate the manifold attracting effect. Through the convergent indexes, the convergent mechanism of the RNBA displays a Hopf bifurcation when approximating the true solutions. Several numerical examples, including the root finding in two-, three-variable and in elliptic-type partial differential equations (PDEs), are examined. Comparisons of numerical results and exact solutions show that the proposed algorithm has good computational efficiency and accuracy simultaneously.

I. INTRODUCTION

Numerical solutions of algebraic equations are widely applied in many fields of engineering and science. In practical engineering problems, numerical methods such as the finite element method (FEM), finite difference method (FDM), boundary element method (BEM), and meshless method are typically used to solve a system of algebraic equations. Regarding the area of computational mechanics, Atluri *et al.* [1-4, 35] have proposed approaches, such as meshless local Petro-Galerkin (MLPG) method and local boundary integral equation (LBIE) method, to obtain solutions for linear algebraic equations of a linear problem and for nonlinear algebraic equations (NAEs) of a nonlinear problem. Until now, a series of approaches [5, 6, 8, 13-15, 18, 21, 22] have been proposed to solve algebraic equations.

Over the past few centuries, many contributions have been made towards finding the solutions of NAEs. Among these methods, Newton's method is a well-known method for finding the roots of a real-valued nonlinear system that can converge quadratically. However, a Newton-like algorithm is sensitive to the initial guess of the solution, and the computations of the Jacobian matrix and its inverse at each iterative step are computationally expensive. Frequently, the requirement of Jacobian matrix inverse causes a failure to search for the solution when a singular Jacobian matrix is encountered. Thus, to overcome these difficulties, the homotopy method was proposed [7] to enhance the convergence of solution by converting the local converging solution to global converging one. The homotopy method finds the solution by constructing an auxiliary function and thus, avoids solving the original problem directly. For the study of homotopy method, researchers such as Davidenko [7], Gorji *et al.* [9], He [10, 11], Liao [19], Wu [34], and Marinca and Herisanu [33] have presented different investigation and applications for different problems. However, many vector-based homotopy methods that determine the inverse of a Jacobian matrix at each time step cannot completely avoid divergence under certain conditions. To overcome this problem of divergence, Liu *et al.* [32] proposed a scalar-based homotopy method, which converts a vector function into a scalar function by taking the squared norm of the vectorial algebraic equations and simultaneously introducing a fictitious time variable. For a scalar-based homotopy method, it does not need to calculate the inverse of the Jacobian matrix and has great numerical stability. However, the main drawback of a scalar-based homotopy method is its slow convergence. That is, the convergence criterion is not easily well satisfied.

In addition to homotopy methods, a novel time integration method called the fictitious time integration method (FTIM)

Paper submitted 11/21/12; revised 05/30/13; accepted 09/12/13. Author for correspondence: Yung-Wei Chen (e-mail: cyw0710@mail.ntou.edu.tw).
Department of Marine Engineering, National Taiwan Ocean University, Keelung, Taiwan, R.O.C.

has been proposed by Liu and Alturi [29]. The FTIM embeds the linear or nonlinear algebraic equations into a system of first-order ordinary differential equations (FOODs) by introducing a time-like or fictitious variable. Liu [23, 24], Liu and Atluri [27, 28, 30], and Liu [25, 26] have demonstrated that FTIM can solve certain classes of problems more effectively than Newton-like methods do. This scheme has the advantage that needs not to calculate the Jacobian matrix or its inverse. With the aim of improving the convergence and a robust implementation, a new time-like function with the incorporation of the FTIM has been proposed by Ku *et al.* [17]. Ku *et al.* [16] then proposed a manifold-based exponentially convergent algorithm (MBECA) for solving NAEs. The MBECA is based on the construction of a space-time manifold by transforming a vector function of algebraic equations into a time-dependent scalar function that employs a fictitious time variable. It uses the concept of plastic flow rate to force the trajectory of the unknown vector onto the manifold by assuming that the direction of the plastic flow rate of the unknown vector is in the gradient direction. The MBECA has the same important features as the FTIM with respect to the ability to solve a large class of problems effectively. Although the MBECA and FTIM have many advantages, both the viscous-damping and time-like function parameters must be determined in advance. In particular, when the magnitudes of these parameters increase, numerical instability occurs. To overcome the above mentioned difficulties, the Residual Norm Based Algorithms (RNBAs) [31] with a construction of a space-time manifold, which is derived from the square norm of the vector function of the algebraic equation, is applied to solve those problems.

The paper is organized as follows. In Section 2, we derive the mathematical formulation of the RNBA and obtain an index of a manifold of nonlinear algebraic equations. In Section 3, we demonstrate five numerical examples, including finding the roots of nonlinear algebraic equations in two, and three variables and comparing our method with the original MBECA in a high-dimensional elliptic-type PDEs. Finally, some important findings and conclusions are summarized in Section 4.

II. THE RESIDUAL-NORM BASED ALGORITHM

1. Construction of Gradient-Flow

Let us consider the following nonlinear algebraic equations (NAEs) in a vector form:

$$\mathbf{F}(\mathbf{x}) = 0. \tag{1}$$

From (1), we can formulate a residual-norm based algorithm (RNBA) from the following manifold [31]:

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

Here, we let **x** be a function of a fictitious time variable *t*. We do not need to specify the function $Q(t)$ a priori, of which $\sqrt{2C/Q(t)}$ is a measure of the residual error in time. At the same time, $Q(t) > 0$ is an increasing function of *t* and *C* is a constant. We let $Q(0) = 1$, and *C* to be obtained by initial condition $\mathbf{x}(0) = \mathbf{x}_0$ with

$$C = \frac{1}{2}\left\|\mathbf{F}(\mathbf{x_0})\right\|^2. \tag{3}$$

In order to determine the evolution equation for **x**, we assume a "normality condition" that

$$\dot{\mathbf{x}} = -\lambda\frac{\partial h}{\partial \mathbf{x}}, \tag{4}$$

where

$$h(\mathbf{x}, t) := \frac{1}{2}Q(t)\left\|\mathbf{F}(\mathbf{x})\right\|^2. \tag{5}$$

A gradient flow can be derived from (2), (4) and (5):

$$\dot{\mathbf{x}} = -q(t)\frac{\left\|\mathbf{F}\right\|^2}{\left\|\mathbf{B}^{\mathrm{T}}\mathbf{F}\right\|^2}\mathbf{B}^{\mathrm{T}}\mathbf{F}, \tag{6}$$

in which **B** is the Jacobian matrix with its *ij*-component being given by $B_{ij} = \partial F_i / \partial x_j$ and

$$q(t) := \frac{\dot{Q}(t)}{2Q(t)}. \tag{7}$$

2. Maintaining the Manifold Property of the Trial Solution

To stay in the manifold defined by (2), we can consider the evolution of F along the path $\mathbf{x}(t)$ by

$$\dot{\mathbf{F}} = \mathbf{B}\dot{\mathbf{x}} = -q(t)\frac{\left\|\mathbf{F}\right\|^2}{\left\|\mathbf{B}^{\mathrm{T}}\mathbf{F}\right\|^2}\mathbf{A}\mathbf{F}, \tag{8}$$

where

$$\mathbf{A} := \mathbf{B}\mathbf{B}^{\mathrm{T}}. \tag{9}$$

Suppose that we use the Euler scheme to integrate (8):

$$\mathbf{F}(t+\Delta t) = \mathbf{F}(t) - \Delta t q(t)\frac{\left\|\mathbf{F}\right\|^2}{\left\|\mathbf{B}^{\mathrm{T}}\mathbf{F}\right\|^2}\mathbf{A}\mathbf{F}, \tag{10}$$

$$\mathbf{x}(t+\Delta t) = \mathbf{x}(t) - \Delta t q(t)\frac{\left\|\mathbf{F}\right\|^2}{\left\|\mathbf{B}^{\mathrm{T}}\mathbf{F}\right\|^2}\mathbf{B}^{\mathrm{T}}\mathbf{F}. \tag{11}$$

Taking the square norms of both sides of (10) and using (2), we can obtain

$$\frac{2C}{Q(t+\Delta t)}=\frac{2C}{Q(t)}-2\big(q(t)\Delta t\big)\frac{2C}{Q(t)}\frac{\mathbf{F}\cdot(\mathbf{AF})}{\|\mathbf{B}^{\mathrm{T}}\mathbf{F}\|^2}$$

$$+\big(q(t)\Delta t\big)^2\frac{2C}{Q(t)}\frac{\|\mathbf{F}\|^2}{\|\mathbf{B}^{\mathrm{T}}\mathbf{F}\|^4}\|\mathbf{AF}\|^2. \tag{12}$$

Thus, we have the following scalar equation:

$$a(\Delta t)^2-2\Delta t+1-\frac{Q(t)}{Q(t+\Delta t)}=0, \tag{13}$$

where

$$a:=q(t)^2\frac{\|\mathbf{F}\|^2\cdot\|\mathbf{AF}\|^2}{\|\mathbf{B}^{\mathrm{T}}\mathbf{F}\|^4}, \tag{14}$$

$$b:=2q(t). \tag{15}$$

Because of the Cauchy-Schwarz inequality:

$$\|\mathbf{B}^T\mathbf{F}\|^2=\mathbf{F}\cdot\mathbf{AF}\le\|\mathbf{F}\|\|\mathbf{AF}\|, \tag{16}$$

we can yield :

$$A0:=\frac{\|\mathbf{F}\|^2\|\mathbf{AF}\|^2}{\|\mathbf{B}^T\mathbf{F}\|^4}\ge 1. \tag{17}$$

Inserting (17) into (13), we can yield

$$A0\,(q\Delta t)^2-2(q\Delta t)+1-S=0, \tag{18}$$

where $S=Q(t)/Q(t+\Delta t)$ and we can take the solution of $q\Delta t$ from (13) to be

$$q\Delta t=\frac{1-\sqrt{1-(1-S)A0}}{A0}\frac{\partial^2\Omega}{\partial v^2},\ \text{if } 1-(1-S)A0\ge 0. \tag{19}$$

3. A Novel Algorithm 1

Let

$$1-(1-S)A0=\gamma^2\ge 0,\ S=1-\frac{1-\gamma^2}{A0}, \tag{20}$$

and thus, we have

$$q\Delta t=\frac{1-\gamma}{A0}. \tag{21}$$

From the Euler method for (11), we can obtain the following residual-norm based algorithm (RNBA):

$$\mathbf{x}(t+\Delta t)=\mathbf{x}(t)-\eta\frac{\|\mathbf{B}^{\mathrm{T}}\mathbf{F}\|^2}{\|\mathbf{AF}\|^2}\mathbf{B}^{\mathrm{T}}\mathbf{F}, \tag{22}$$

where

$$\eta=1-\gamma. \tag{23}$$

Here, $0\le\gamma<1$ is a parameter choosing by user and $0\le\eta\le 1$ is a weighting factor. As $\gamma=0$, it is interesting that in (22) no parameter and no Δt are required. That is, it guarantees the RNBA to be absolutely convergent to the true solution. Under the above conditions, we can prove that the new algorithm RNBA satisfies

$$\frac{\mathbf{F}(\mathrm{t}+\Delta t)}{\mathbf{F}(\mathrm{t})}=\sqrt{S},\ \text{if } 1-(1-S)A0=\gamma^2\ge 0, \tag{24}$$

$$\frac{\mathbf{F}(\mathrm{t}+\Delta t)}{\mathbf{F}(\mathrm{t})}=\sqrt{1-A0^{-1}},\ \text{if } 1-(1-S)A0<0. \tag{25}$$

which means that the residual error is absolutely decreased.

4. A Novel Algorithm 2

According to (22), we can understand that $\|\mathbf{B}^{\mathrm{T}}\mathbf{F}\|^2\,\mathbf{B}^{\mathrm{T}}\mathbf{F}$ / $\|\mathbf{AF}\|^2$ is a regularized gradient vector. The weighting factor η can be placed to speed up the convergence speed. Here, we introduce the concept of the group-preserving scheme (GPS) into RNBA. The GPS first proposed by Liu [20] is a numerical scheme with a space-time manifold, which maintains the manifold property and changes the weighting factor. Then, the GPS can be presented as follows:

$$\mathbf{x}(t+\Delta t)=\frac{b(t)\|\mathbf{x}(t)\|\ \|\mathbf{F}(t)\|+(a(t)-1)\mathbf{F}(t)\cdot\mathbf{x}(t)}{\|\mathbf{F}(t)\|}$$

$$=\mathbf{x}(t)+\eta(t)\mathbf{F}(t), \tag{26}$$

$$a(t):=\cosh\Big(\frac{\Delta t\|\mathbf{F}(t)\|}{\|\mathbf{x}(t)\|}\Big), \tag{27}$$

$$b(t):=\sinh\Big(\frac{\Delta t\|\mathbf{F}(t)\|}{\|\mathbf{x}(t)\|}\Big). \tag{28}$$

Thus, under the above conditions, we can use exponential function to replace with $a(t)$ and $b(t)$, and develop an automatically adaptive numerical method as follows:

$$\eta(t)=(1-\alpha)\exp\left(-\frac{\beta\|\mathbf{F}(t)\|}{\|\mathbf{x}(t)\|}\right). \tag{29}$$

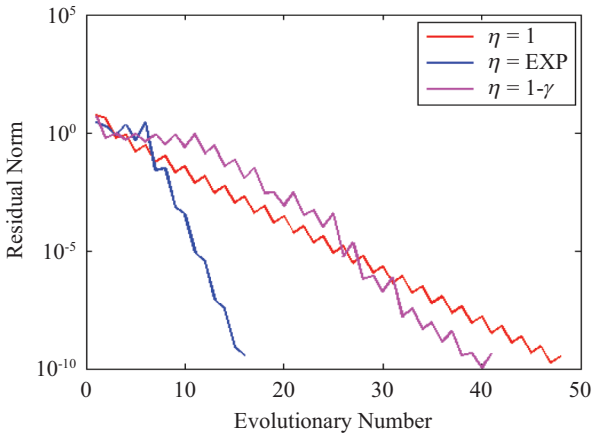

Fig. 1. A nonlinear equation (30) solved by the RNBA with different weighting factors at the initial value of (50, -30).

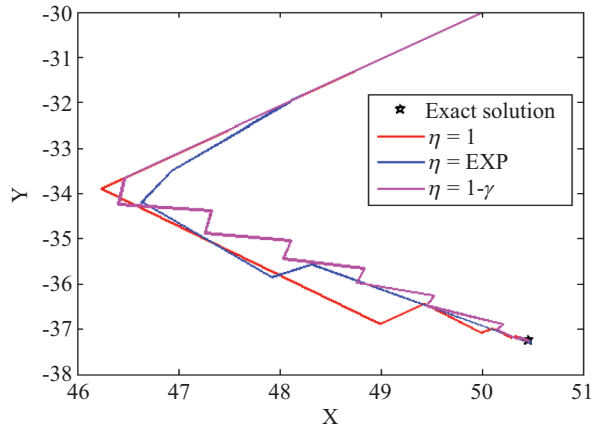

Fig. 2. A nonlinear equation (30) solved by the RNBA comparing the iterative path with different weighting factors.

Here, $0 \le \alpha < 1$ is the range of the fictitious time weighting factor and β is a fictitious time step size chosen by the user. When the solutions are obtained, the norm of **F** will become zero.

In the following we will give some numerical tests of the Residual Norm Based Algorithms (RNBAs): the RNBAs with $\gamma = 0$, the RNBA with $1 - \gamma$, and the RNBAs associated with the GPS, respectively.

III. NUMERICAL EXAMPLES

Example 1

Let's consider a system of two NAEs with two variables that were previously studied by Hirsch and Smale [12]:

$$F_1(x,y) = x^3 - 3x^2y + a_1(2x^2 + xy) + b_1y^2 + c_1x + a_2y = 0,$$

$$F_2(x,y) = 3x^2y - y^3 - a_1(4xy - y^2) + b_2x^2 + c_2 = 0, \quad (30)$$

where $(a_1, b_1, c_1, a_2, b_2, c_2) = (25, 1, 2, 3, 4, 5)$.

For this problem, Liu and Atluri [29] solved (30) by using the FTIM, and Liu and Atluri [30] then found four solutions. To examine the characteristics of the RNBA, we resolved the same problem with the following parameters: $\gamma = 0$, $\gamma = 0.08$, $\alpha = 0$, and $\beta = 0.005$. Starting from an initial value of $(x, y) = (50, -30)$, we solved this problem using the RNBA with different weighting factors under a convergence criterion of $\varepsilon = 1 \times 10^{-10}$. We compared the residual norms and iterative paths of the RNBA with algorithms 1 and 2, and the results are shown in Figs. 1 and 2 respectively, of which the red, pink and blue lines denote the RNBA by algorithm 1 with $\gamma = 0$, algorithm 1 with $\gamma = 0.06$ and algorithm 2 with $\alpha = 0$ and $\beta = 0.0005$, respectively. The numerical results are very close to the true solution with the residual on the order of 10^{-12}, and the numerical solution path of the RNBA converges to the solution $(x, y) = (50.46504, -37.2634)$ when the evolution numbers are at 49, 42 and 17 steps. From the path of the numerical results, shown in Fig. 2, we can observe that algorithm 2 forces the path to depart from the manifold and searches the optimal path to approach the true solution. Also, we show the residual norm, *A0*, *S*, and η of the RNBA by algorithms 1 and 2 in Figs. 3, 4 and 5, respectively. As shown in Fig. 3, we can observe that *A0* and *S* maintain a moderate value when using algorithm 1 with $\gamma = 0$ searching along the manifold. We can clearly see that *A0* and *S* of algorithm 1 with $\gamma = 0.06$, as shown in Fig. 4, reveal that there is oscillation when the searching direction is attracted by the manifold. Again, as shown in Fig. 5, *A0* maintains a very small value and η returns to one by algorithm 2 when the gradient direction is found.

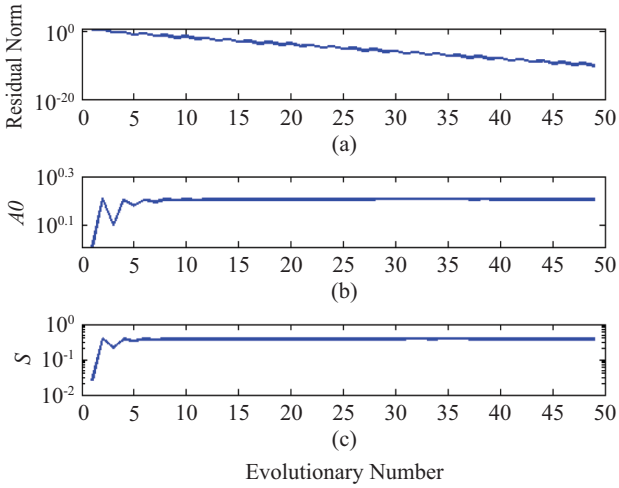

Fig. 3. A nonlinear equation (30) solved by the RNBA by algorithm 1 with $\gamma = 0$ showing (a) the residual norm, (b) *A0* and (c) *S*.

In this problem, there is another solution given by $(x, y) = (36.045402, 36.80750808)$. For this solution, the parameters are given by $\gamma = 0$, $\gamma = 0.06$, $\varepsilon = 1 \times 10^{-10}$, $\alpha = 0$ and $\beta = 0.0043$, and the initial value is $(x, y) = (40, 20)$. We compare the residual norms and iteration paths of the RNBA by algorithms 1 and 2. The results are shown in Figs. 6 and 7, respectively, of

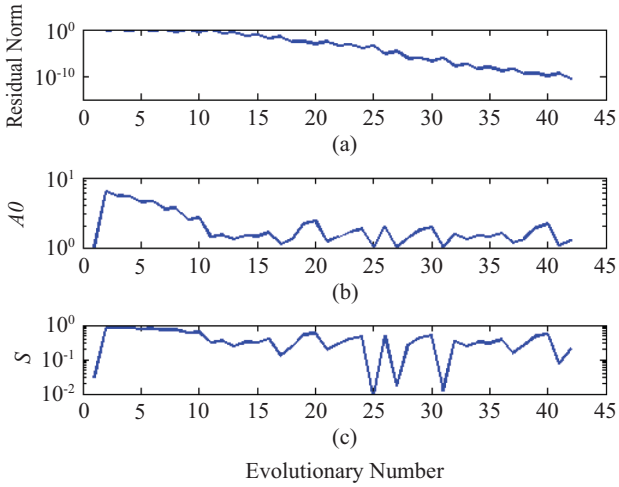

Fig. 4. A nonlinear equation (30) solved by the RNBA by algorithm 1 with γ = 0.06 showing (a) the residual norm, (b) *A0* and (c) *S*.

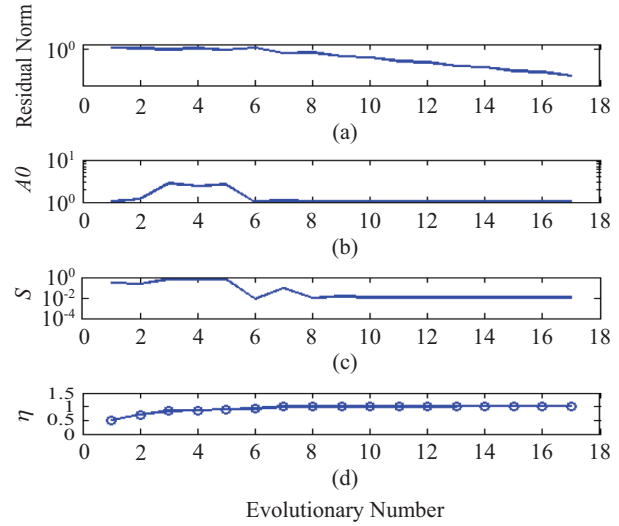

Fig. 5. A nonlinear equation (30) solved by the RNBA by algorithm 2 with α = 0 and β = 0.0005 showing (a) the residual norm, (b) *A0*, (c) *S* and (d) η.

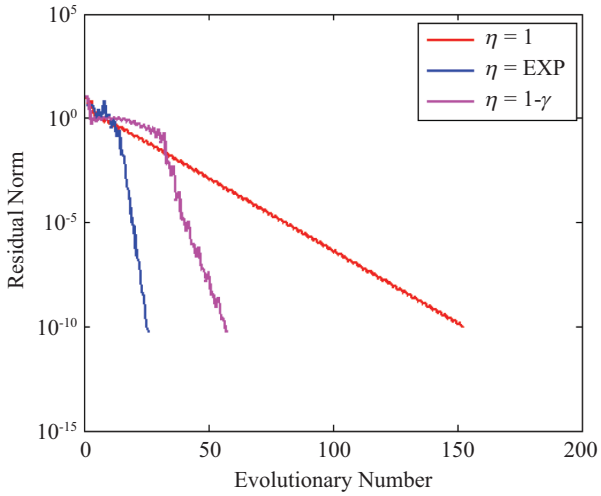

Fig. 6. A nonlinear equation (30) solved by the RNBA with different weighting factors at the initial value of (40, 20).

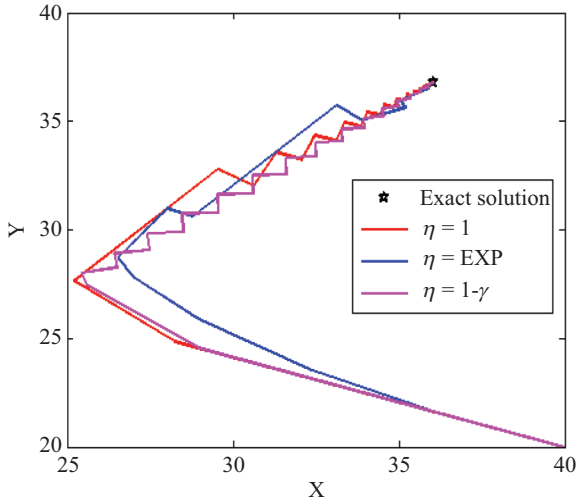

Fig. 7. A nonlinear equation (30) solved by the RNBA comparing the iterative path with different weighting factors.

which the red, pink and blue lines denote the RNBA by algorithm 1 with γ = 0, algorithm 1 with γ = 0.06 and algorithm 2 with α = 0 and β = 0.0043, respectively. We can observe that the numerical results are very close to the true solution with the residual on the order of 10^{-10}, and the result is obtained at 152, 57 and 26 steps. We also show the residual norm, *A0*, *S*, and η of the RNBA by algorithms 1 and 2 in Figs. 8, 9 and 10, respectively. In this case, we can clearly observe that the iterative path zigzags when the searching path is attracted by the manifold. With the parameters of $\varepsilon = 1 \times 10^{-10}$, γ = 0, γ = 0.08, α = 0 and β = 0.005, and starting from an initial value of (x, y) = (10, 10), we can find the fifth root (x, y) = (1.635972, 13.847665). The results of the residual norm and iterative paths are shown in Figs. 11 and 12, respectively, in which the red, pink and blue lines denote the RNBA by algorithm 1 with γ = 0, algorithm 1 with γ = 0.08 and algorithm 2 with α = 0 and β = 0.005, respectively; the solution is obtained at 1768, 244 and 64 steps. In addition, we show the residual norms, *A0*, *S*, and η of the RNBA by algorithms 1 and 2 in Figs. 13-15. As mentioned above, the RNBA becomes less computationally efficient and a Hopf bifurcation occurs when the iteration path is very close to the manifold.

Example 2

In this example, we study the following system of two algebraic equations:

$$F_1(x_1, x_2) = x_1^2 + x_2^2 - 2 = 0,$$

$$F_2(x_1, x_2) = e^{(x_1-1)} + x_2^2 - 2 = 0. \quad (31)$$

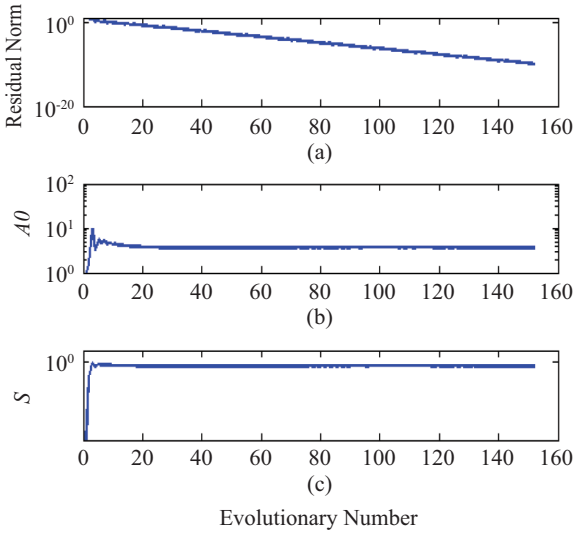

Fig. 8. A nonlinear equation (30) solved by the RNBA by algorithm 1 with γ = 0 showing (a) the residual norm, (b) *A0* and (c) *S*.

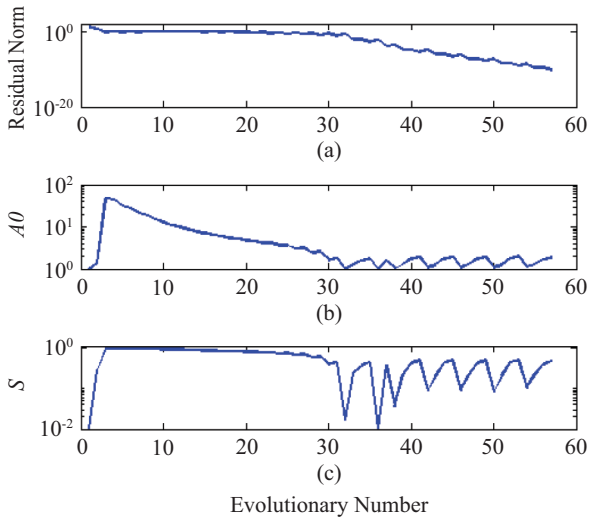

Fig. 9. A nonlinear equation (30) solved by the RNBA by algorithm 1 with γ = 0.06 showing (a) the residual norm, (b) *A0* and (c) *S*.

The Jacobian matrix of (31) can then be expressed as follows:

$$\mathbf{B} = \begin{bmatrix} 2x_1 & 2x_2 \\ e^{(x_1-1)} & 2x_2 \end{bmatrix}. \tag{32}$$

This example has been discussed by Kuo, Yeih, and Liu [16] using the scalar Newton-homotopy continuation method. The solution search easily fails at the stagnation point (x_1, x_2) = (3.5192, 0) because the Jacobian matrix **B** derived from the target functions is singular. We use this test to examine the numerical stability of the RNBA. Setting the parameters as $\gamma = 0$, $\gamma = 0.04$, $\alpha = 0$, $\beta = 0.152$, $\varepsilon = 1 \times 10^{-10}$ and starting from an initial value of (x_1, x_2) = (3, 5), we compare the residual

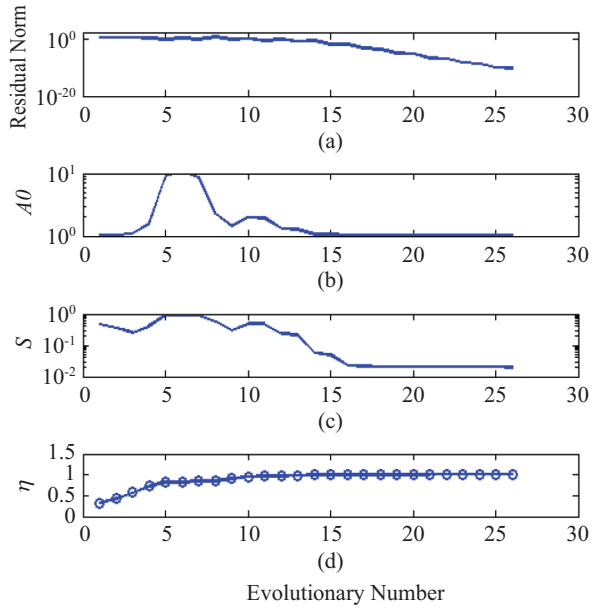

Fig. 10. A nonlinear equation (30) solved by the RNBA by algorithm 2 with α = 0 and β = 0.005 showing (a) the residual norm, (b) *A0*, (c) *S* and (d) η.

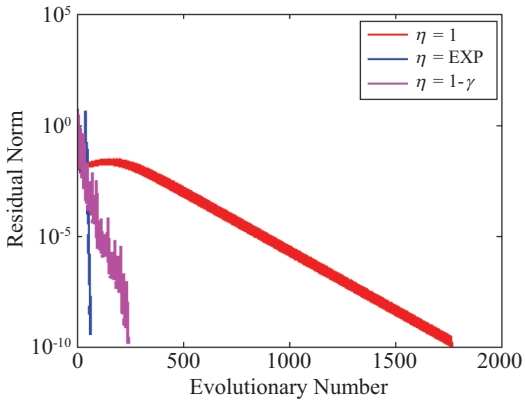

Fig. 11. A nonlinear equation (30) solved by the RNBA with different weighting factors at the initial value of (10, 10).

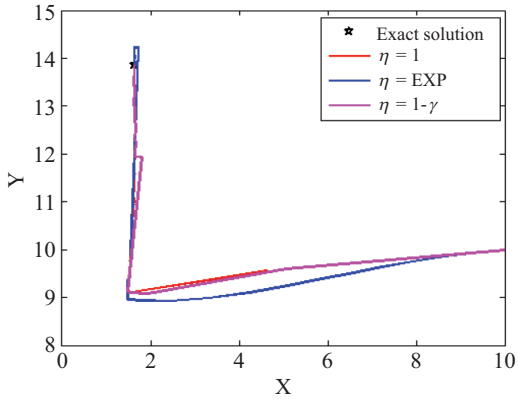

Fig. 12. A nonlinear equation (30) solved by the RNBA, comparing the iterative path with different weighting factors.

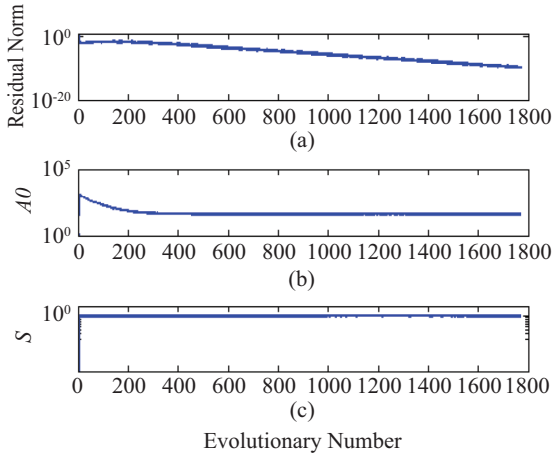

Fig. 13. A nonlinear equation (30) solved by the RNBA with γ = 0 showing (a) the residual norm, (b) *A0* and (c) *S*.

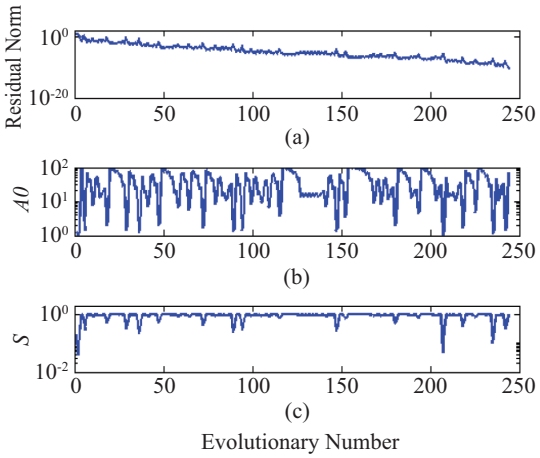

Fig. 14. A nonlinear equation (30) solved by the RNBA with γ = 0.06 showing (a) the residual norm, (b) *A0* and (c) *S*.

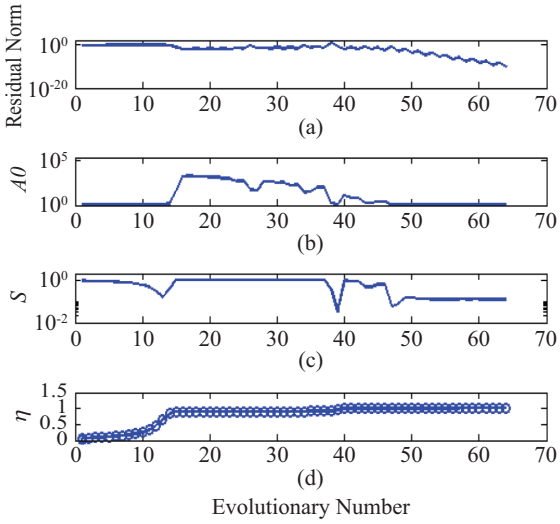

Fig. 15. A nonlinear equation (30) solved by the RNBA by algorithm 2 with α = 0 and β = 0.05 showing (a) the residual norm, (b) *A0*, (c) *S* and (d) η.

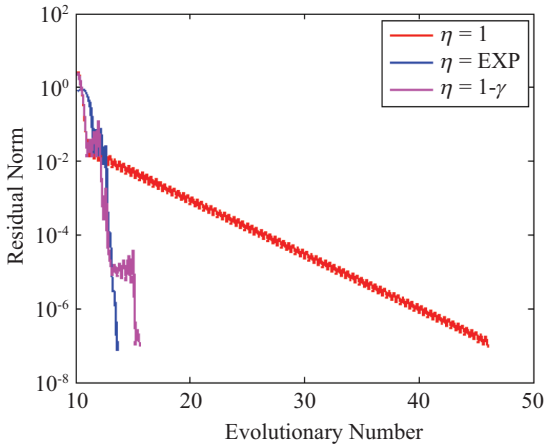

Fig. 16. A nonlinear equation (31) solved by the RNBA with different weighting factors.

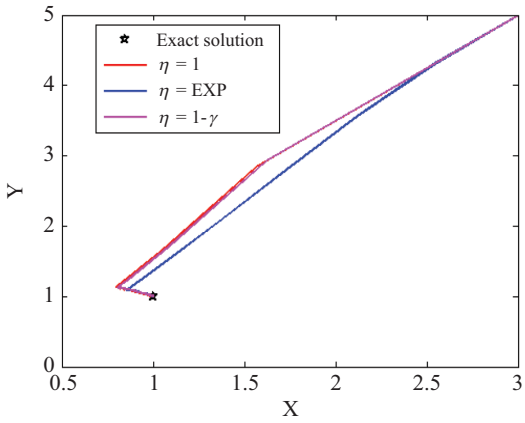

Fig. 17. A nonlinear equation (31) solved by the RNBA comparing the iterative paths with different weighting factors.

norm of the RNBA by algorithms 1 and 2, respectively; the results are shown in Figs. 16 and 17, in which the red, pink and blue lines denote the RNBA by algorithm 1 with γ = 0, algorithm 1 with γ = 0.04 and algorithm 2 with α = 0 and β = 0.152, respectively. The numerical solution is very close to the true solution with the residual error on the order of 10^{-7} as shown in Fig. 16, and the iterative path of the RNBA converges to the solution (x_1, x_2) = (1, 1) in 180 steps for algorithm 1 with γ = 0, 28 steps for algorithm 1 with γ = 0.04, and 18 steps for algorithm 2, respectively. In addition, we show the residual norm, *A0*, *S*, and η in Figs. 18-20.

Example 3

In this example, we study the following system of three algebraic equations:

$$F_1(x,y,z) = x + y + z - 3 = 0,$$

$$F_2(x,y,z) = xy + 2y^2 + 4z^2 - 7 = 0,$$

$$F_3(x,y,z) = x^8 + y^4 + z^9 - 3 = 0. \qquad (33)$$

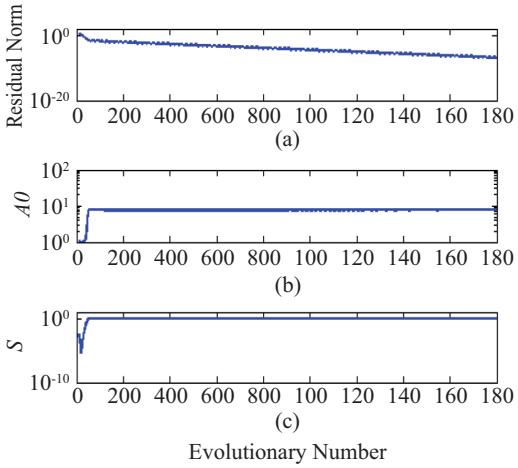

Fig. 18. A nonlinear equation (31) solved by the RNBA by algorithm 1 with $\gamma = 0$ showing (a) the residual norm, (b) *A0* and (c) *S*.

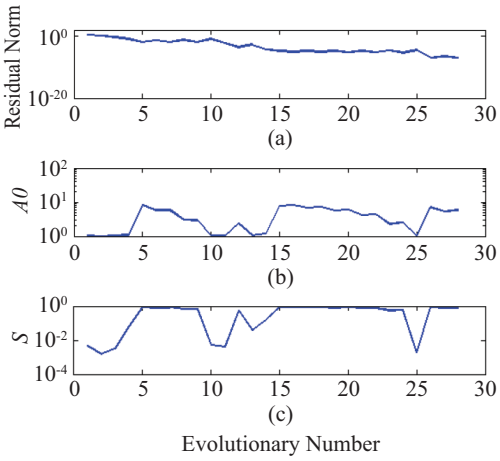

Fig. 19. A nonlinear equation (31) solved by the RNBA by algorithm 1 with $\gamma = 0.04$ showing (a) the residual norm, (b) *A0* and (c) *S*.

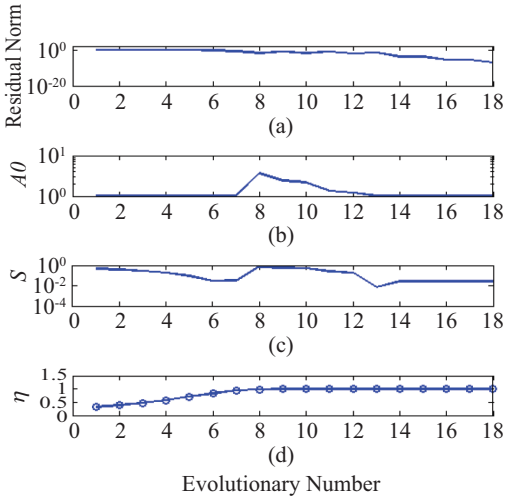

Fig. 20. A nonlinear equation (31) solved by the RNBA by algorithm 2 with $\alpha = 0$ and $\beta = 0.152$ showing (a) the residual norm, (b) *A0*, (c) *S* and (d) η.

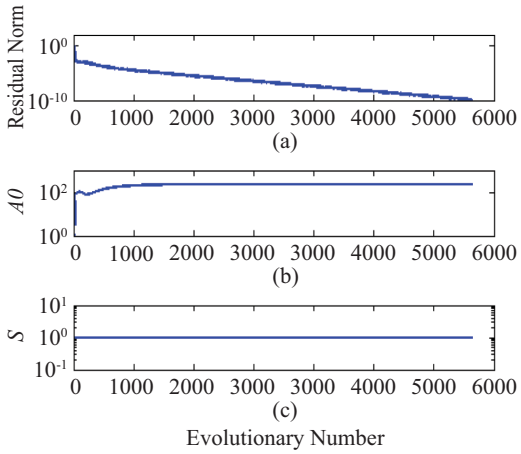

Fig. 21. A nonlinear equation (33) solved by the RNBA by algorithm 1 with $\gamma = 0$ showing (a) the residual norm, (b) *A0* and (c) *S*.

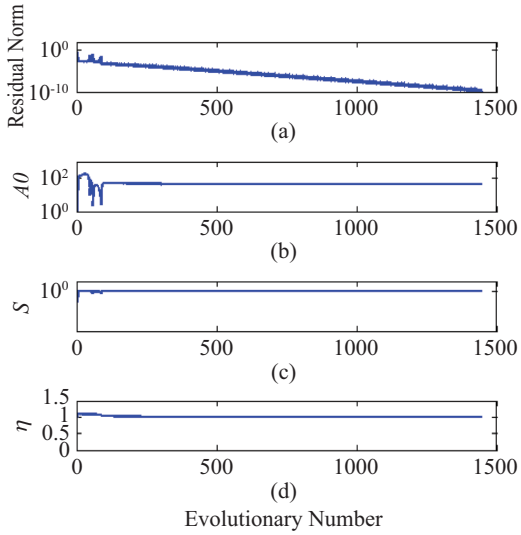

Fig. 22. A nonlinear equation (33) solved by the RNBA with $\alpha = 0$ and $\beta = 0.1$ showing (a) the residual norm, (b) A0, (c) S and (d) η.

To observe the relationship between the manifold and the parameters, we set the parameters $\varepsilon = 1 \times 10^{-10}$ and $\gamma = 0$ and start from $(x, y, z) = (0.5, 0.6, 0.6)$ to obtain the solution $(x, y, z) = (1, 1, 1)$. The residual norm, *A0* and *S* are shown in Figs. 21(a), 21(b), and 21(c), respectively. In Fig. 21(a), the solution is obtained in 6640 steps, and the residual norm is on the order of 10^{-11}. In Fig. 21(b), we can clearly see that a moderate value of *A0* exists in the system, which denotes the system limit of the manifold. That is, the limit of the manifold needs to be broken through to improve the computational efficiency and accuracy of the numerical scheme, and, at the same time, *A0* and *S* must remain as small values. To demonstrate how to break through the attracting effect of the manifold, we use algorithm 2 with $\alpha = 0$ and $\beta = 0.1$, and the numerical result is shown in Fig. 22. In 1447 steps, the solutions of system are obtained with the same residual norm. We also show the

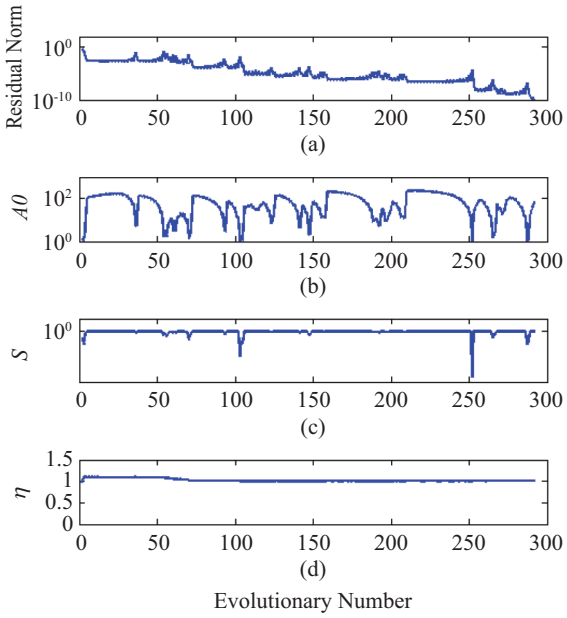

Fig. 23. A nonlinear equation (33) solved by the RNBA with α = 0.016 and β = 0.1 showing (a) the residual norm, (b) *A0*, (c) *S* and (d) η.

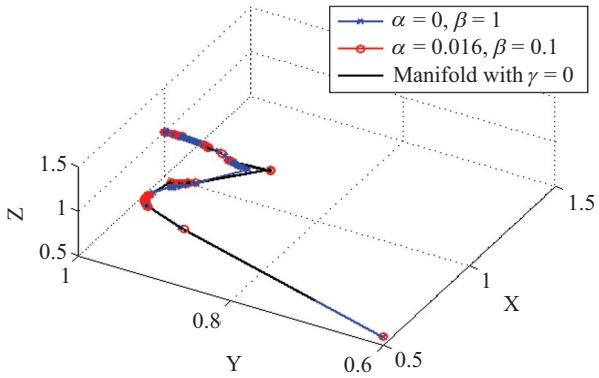

Fig. 24. Comparison of the iterative path with different weighting factors.

residual norm, *A0*, *S*, and η in Fig. 22. We further find that by using a large fictitious time step as shown in Fig. 22(b), the numerical solution quickly approaches to the true one. However, fictitious time step is not the sole factor that controls whether the convergence path can depart from the manifold.

With the parameters of α = 0.016 and β = 0.1 and the same convergence criterion, the residual norm, *A0*, *S*, and η are shown in Fig. 23. As shown in Fig. 23(a), only within 292 steps, the result of the RNBA converges to the solution (*x*, *y*, *z*) = (0.999999, 0.999999, 1), with the residual norm on the order of 10^{-11}. We can see that the RNBA adopts an oscillatory fashion to depart from the manifold and to find an optimal evolution path. In Fig. 24, we compare the iterative paths of α = 0.016, α = 0 and γ = 0. We can clearly observe that when a large fictitious time step with α = 0 and β = 0.1, is used, the path departs from the manifold only in the first few steps.

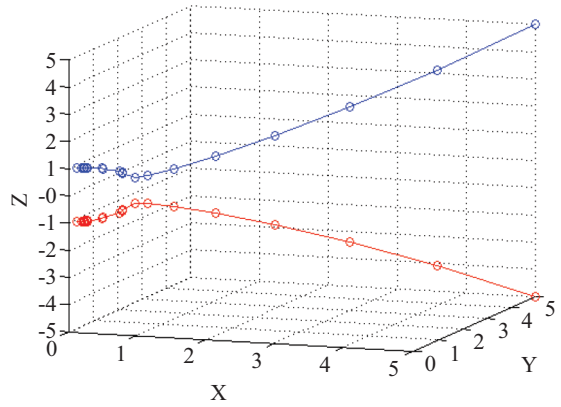

Fig. 25. Two solutions of equation (34) solved by the RNBA comparing the iterative paths from different initial conditions.

However, it is not real and thus, cannot break through the manifold. On the contrary, the path of the RNBA with α = 0.016 and β = 0.1 is parallel to the manifold, which means that the optimal path is the maximal gradient direction.

Example 4

This example under investigation is a system of two NAEs in three variables as follows:

$$F_1(x,y,z) = x^2 + y^2 + z^2 - 1 = 0,$$

$$F_2(x,y,z) = \frac{x^2}{4} + \frac{y^2}{4} + z^2 - 1 = 0. \qquad (34)$$

In this problem, number of equations is less than that of variables for the underdetermined nonlinear system which raises the difficulty of finding solutions by numerical schemes. With the parameters of $\varepsilon = 1 \times 10^{-7}$, α = 0 and β = 0.01 and starting from (*x*, *y*, *z*) = (5, 5, 5) and (5, 5, -5), we can easily find that (*x*, *y*, *z*) = (0, 0, 1) and (*x*, *y*, *z*) = (0, 0, -1), as shown in Fig. 25, in which the solutions satisfy (34) under 179 steps. It can be seen that the RNBA approaches to the solution very rapidly and that the solution path is an optimisation of the local solution. Also, we show the residual norm, *A0*, *S*, and η in Fig. 26. We can see that this scheme masterfully avoids being constrained by the manifold and finds the numerical solution with the residual norm on the order of 10^{-7} when both solutions are neighbours.

Example 5

For this example, we use the RNBA associated with the NFDM to solve the elliptic boundary value problem. For this problem, we need to obtain multi-variables in a higher-dimensional space, and at the same time, the residual is not equal to zero. A two-dimensional arbitrary domain with Dirichlet boundary conditions is further considered to test the accuracy and stability of this algorithm.

The computational domain is plotted in Fig. 27, and the

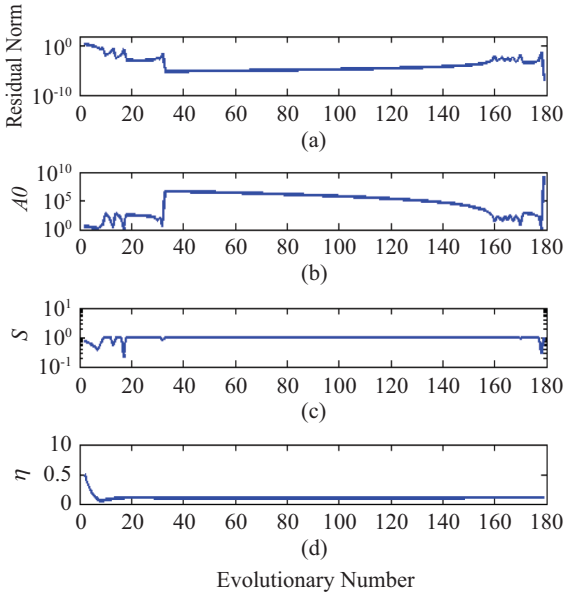

Fig. 26. A nonlinear equation (34) solved by the RNBA with α = 0 and β = 0.01 showing (a) the residual norm, (b) *A0*, (c) *S* and (d) η.

contour shape of this problem is given by

$$\Omega=\{(x,y)\,|\,x=\rho\cos\theta+1.5,\ y=\rho\sin\theta+1.5,\ 0\le\theta\le 2\pi\}, \tag{35}$$

where

$$\rho=\left(\cos(3\theta)+\sqrt{2-\sin^2(3\theta)}\right)^{\frac{1}{3}}. \tag{36}$$

We then consider the analytical solution

$$u(x,y)=e^x\cos y+20 \tag{37}$$

of the Laplace equation

$$\nabla^2 u=0,\ (x,y)\in\Omega. \tag{38}$$

We apply a finite difference procedure to the Laplace operator and discretize the domain with the finite difference method with a uniform grid. Then, (38) can be rewritten as follows:

$$\frac{u_{i+1,j}-2u_{i,j}+u_{i-1,j}}{(\Delta x)^2}+\frac{u_{i,j+1}-2u_{i,j}+u_{i,j-1}}{(\Delta y)^2}=0,$$

$$i,\ j=2,\ 3,\ 4,\ldots,\ m-1. \tag{39}$$

In order to conveniently deal with an arbitrary computational domain using the FDM, we introduce a fictitious

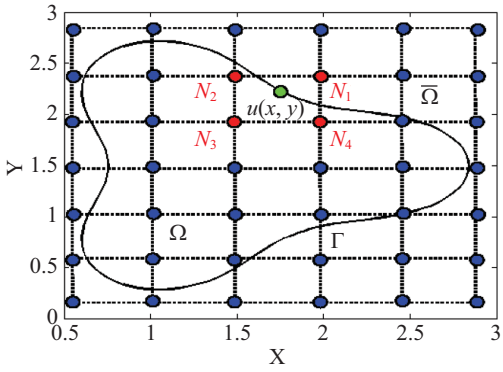

Fig. 27. The shape of the calculation domain.

equidistant rectangle to enclose the problem domain Ω as shown in Fig. 27.

Suppose that the rectangle is given by $\bar{\Omega}:=[-a_x,\ a_x]\times[-b_y,\ b_y]$ such that a_x and b_y can cover the maximum length of the problem domain along the *x*-axis and *y*-axis, respectively. Next, we discretize the rectangle $\bar{\Omega}$ by a uniform grid, with $\Delta x = 2a_x/(m-1)$ and $\Delta y = 2b_y/(m-1)$ in the *x*- and *y*-directions, respectively. At the same time, we let $u_{i,j}(t) = u(x_i, y_j, t)$ be a numerical value of *u* at the grid point $(x_i, y_j) \in \bar{\Omega}$, which varies with fictitious time *t*, where $x_i = (i-1)\Delta x$ and $y_i = (i-1)\Delta y$.

To overcome the difficulty of imposing boundary condition on arbitrary boundary under FDM framework, we introduce the concept of a shape function to address the boundary condition more easily. Let ξ and Ψ be the local coordinates in the *x*- and *y*-directions in the grid space $\bar{\Omega}$, respectively. We then define ξ and Ψ as follows:

$$\xi=2\left(\frac{x-x_{i,j}}{\Delta x}\right)-1,\ \ i,j=1,2,3,\ldots,m, \tag{40}$$

$$\Psi=2\left(\frac{y-y_{i,j}}{\Delta y}\right)-1,\ \ i,j=1,2,3,\ldots,m, \tag{41}$$

where (x, y) are the global coordinates on the boundary and $(x_{i,j}, y_{i,j})$ denote the coordinates of grid point $\bar{\Omega}$. By inserting (40) and (41) into the shape function, the weighting coefficient of the boundary condition can be expressed as follows:

$$N_1=\frac{1}{4}(1+\xi)(1+\Psi),$$

$$N_2=\frac{1}{4}(1-\xi)(1+\Psi),$$

$$N_3=\frac{1}{4}(1-\xi)(1-\Psi),$$

$$N_4=\frac{1}{4}(1+\xi)(1-\Psi), \tag{42}$$

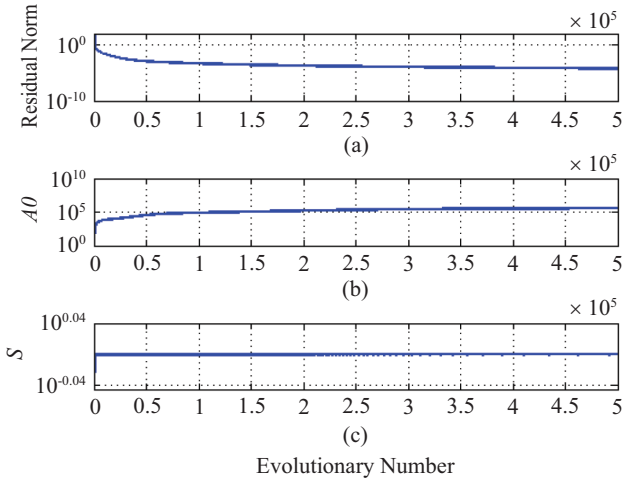

Fig. 28. Example 5 solved by the RNBA by algorithm 1 with γ = 0 showing (a) the residual norm, (b) *A0* and (c) *S*.

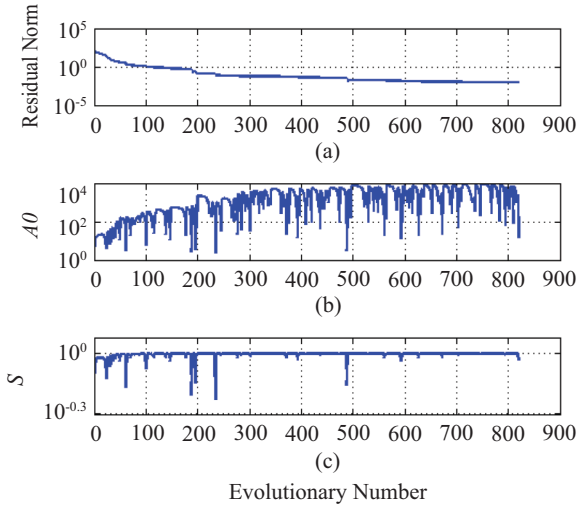

Fig. 29. Example 5 solved by the RNBA by algorithm 1 with γ = 0.08 showing (a) the residual norm, (b) *A0* and (c) *S*.

where N_1, N_2, N_3 and N_4 denote the bi-linear FEM shape functions. When the boundary node is located in the element, a boundary condition and (39) and (42) are introduced to obtain the following algebraic equations to enforce the boundary condition:

$$\sum_{i=1}^{4} N_i^{b_1} u_{b_1} - H_{b_1} = 0,$$

$$\sum_{i=1}^{4} N_i^{b_2} u_{b_2} - H_{b_2} = 0,$$

$$\vdots$$

$$\sum_{i=1}^{4} N_i^{b_n} u_{b_n} - H_{b_n} = 0, \tag{43}$$

where b_1, b_2 … b_n are the boundary nodes, and N^{b_1}, N^{b_2} … N^{b_n} and $H_{b_1}, H_{b_2} \ldots H_{b_n}$ denote the bi-linear Finite Element Method (FEM) shape functions associated with the boundary points and boundary conditions, respectively. Finally, (39) and (43) can be written as an algebraic equation as (1). To obtain the numerical solution, one can solve it by the RNBA.

To observe the relation between accuracy and the manifold factor *A0*, we use different convergence criteria to test this problem. First, we fix the following parameters: $a_x = b_y = 3$, $m = 20$, $n = 160$, $\mathbf{x_0} = \mathbf{1}$, and $\varepsilon = 8 \times 10^{-6}$, which denote the length of fictitious rectangle in the *x*-direction and *y*-direction, the number of fictitious grids in the *x*-direction and *y*-direction, the total number of boundary points, the vector of initial guess, and the convergent criterion, respectively. With the problem defined above, the manifold of the RNBA with $\gamma = 0$ is shown in Fig. 28. We find that the existence of a manifold for the RNBA with $\gamma = 0$ in a high-dimensional space will not easily converge to the true solution, and the evolution numbers, even

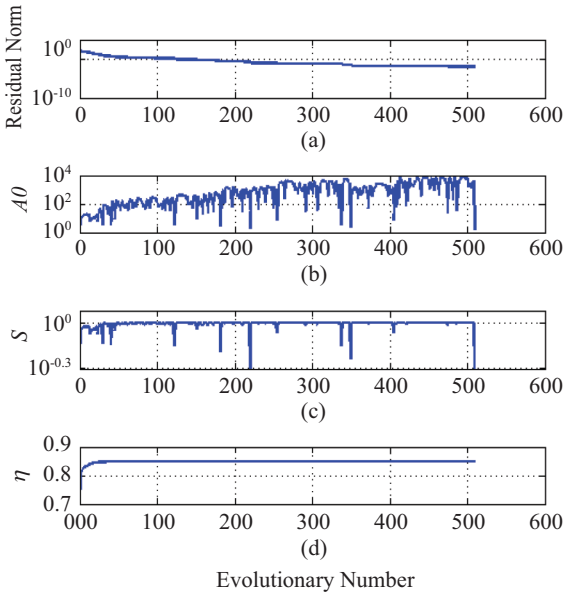

Fig. 30. A nonlinear equation solved by the RNBA with α = 0.15 and β = 0.01 showing (a) the residual norm, (b) *A0*, (c) *S* and (d) η.

up to 5×10^5, cannot satisfy the convergence criterion. Also, we observe that the trend of *A0* will show anti-symmetry with the residual norm in a high-dimensional problem. That is, this algorithm may closely approach the solutions, but one cannot further improve the accuracy with more iteration steps when the iterative process is constrained by the manifold. To observe how the variation of the RNBA can lead to departure from the manifold in high-dimensional space, we fix the convergence criterion with $\varepsilon = 8 \times 10^{-3}$ and use algorithm 1 with γ = 0.08 and algorithm 2 with α = 0.15 and β = 0.01. The residual norm, *A0*, and *S* of the RNBA by algorithm 1 and 2 are shown in Figs. 29 and 30, respectively. We can see that the residual norms of algorithm 1 and algorithm 2 decrease

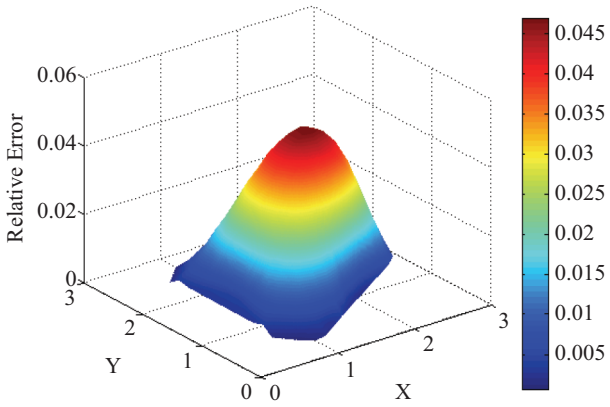

Fig. 31. Example 5 displaying the relative error of the RNBA by algorithm 1.

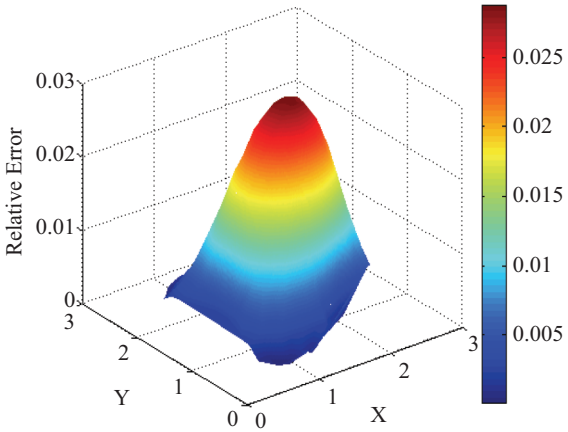

Fig. 32. Example 5 displaying the relative error of the RNBA by algorithm 2.

very quickly in 822 steps for algorithm 1 and 510 steps for algorithm 2. Furthermore, we can observe that the convergence processes of algorithm 1with $A0$ = 18 and S = 0.94488 and of algorithm 2 with $A0$ = 1.6 and S = 0.4197 are not attracted by the manifold, as shown in Figs. 29(b) and 30(b), respectively; this phenomenon is called intermittent chaos. Both algorithms 1 and 2 give accurate numerical results. In addition, as shown in Figs. 31 and 32, the maximum errors are 4.68038×10^{-2} and 2.86859×10^{-2}, respectively. We can then observe the effect of the RNBA with $\varepsilon = 8 \times 10^{-6}$ and compare it with the MBECA when an attractable phenomenon occurs between the manifold and true solution.

With the parameters $\mathbf{x_0} = \mathbf{1}$, $\gamma = 0.08$, $\alpha = 0.1$ and $\beta = 0.01$, we compare the convergent rates of the RNBA with different weighting factors with the MBECA; the convergence plot is shown in Fig. 33, in which the black, red, pink and blue lines denote the RNBA by algorithm 1, RNBA by algorithm 2, RNBA with $\gamma = 0$ and MBECA, respectively. We can see that the residual norms from the RNBA by algorithms 1 and 2 decrease very quickly and that the MBECA exhibits many irregular bursts. Both the MBECA and RNBA give rather

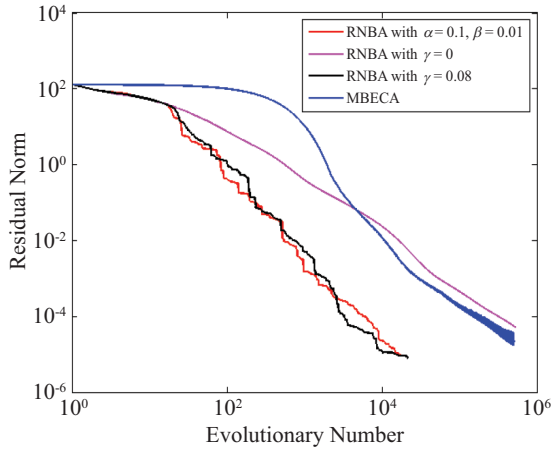

Fig. 33. The convergence plot with the black, pink, red, and blue lines denoting the RNBA by algorithm 1 with γ = 0.08, the RNBA by algorithm 1 with γ = 0, the RNBA by algorithm 2 with α = 0.1 and β = 0.01, and the MBECA, respectively.

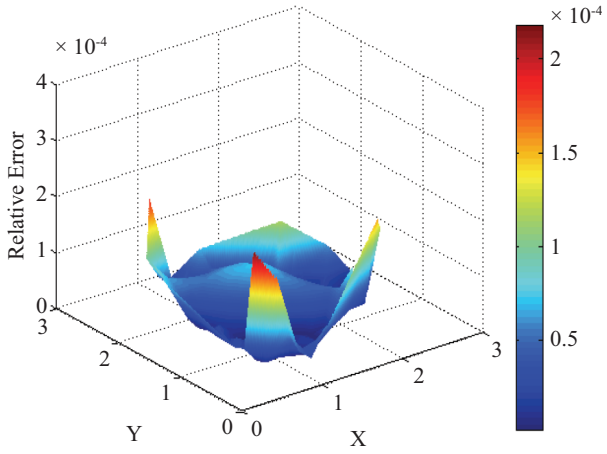

Fig. 34. Example 5 displaying the relative error of the RNBA by algorithm 1.

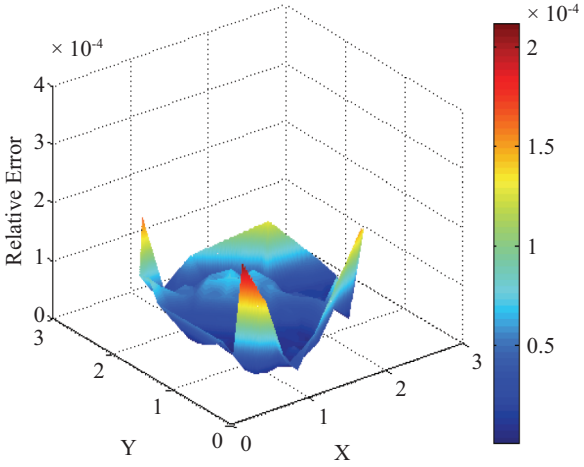

Fig. 35. Example 5 displaying the relative error of the RNBA by algorithm 2.

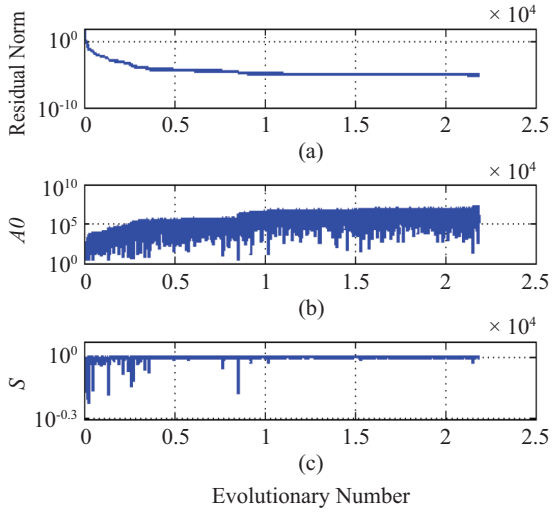

Fig. 36. Example 5 solved by the RNBA by algorithm 1 with γ = 0.08, showing (a) the residual norm, (b) *A0* and (c) *S*.

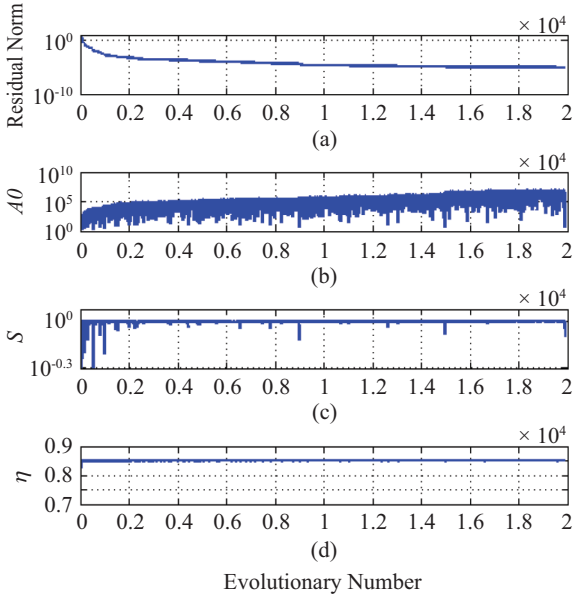

Fig. 37. Example 5 solved by the RNBA with α = 0.1 and β = 0.01 showing (a) the residual norm, (b) *A0*, (c) *S* and (d) η.

accurate numerical results with maximum errors of 4.35×10^{-4} for the MBECA, 2.22×10^{-4} for the RNBA by algorithm 1, and 2.08×10^{-4} for the RNBA by algorithm 2. In Figs. 34 and 35, we only show the numerical errors for the RNBA by algorithm 1 and 2. In addition, the residual norm, *A0*, and *S* of the RNBA by algorithm 1 and 2 are shown in Figs. 36 and 37, respectively. We can see that the residual norms of algorithm 1 and algorithm 2 satisfy the convergence criterion in 2.1865×10^4 steps for algorithm 1 and 1.9897×10^4 steps for algorithm 2. Furthermore, we can observe that $A0 = 4.175 \times 10^6$ and $S = 0.9999$ for algorithm 1 and $A0 = 4.9088$ and $S = 0.79832$ for algorithm 2, which are not attracted by the manifold, as

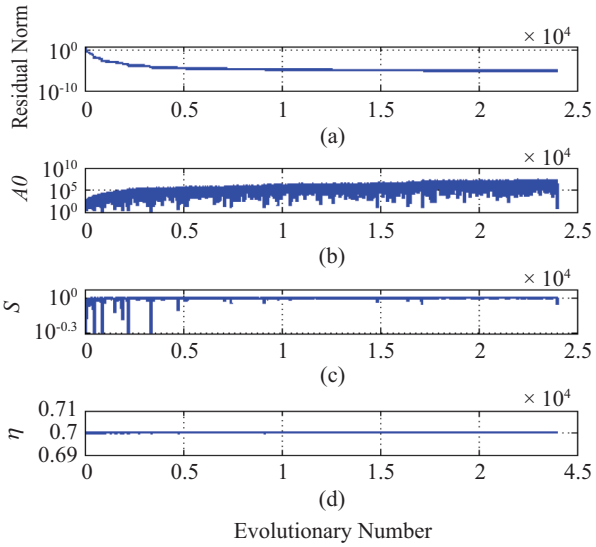

Fig. 38. Example 5 solved by the RNBA with α = 0.3 and β = 0.001 showing (a) the residual norm, (b) *A0*, (c) *S* and (d) η.

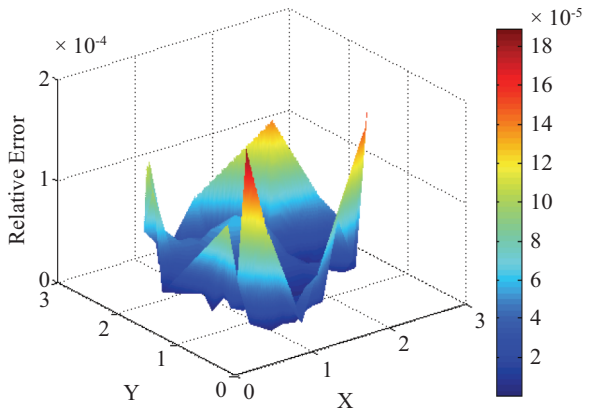

Fig. 39. Example 5 displaying the relative error of the RNBA by algorithm 2.

shown in Figs. 36(b) and 37(b). Finally, to obtain a more accurate solution, we can fix the small values α = 0.3 and β = 0.001 to approach the true solution, and the residual norm, *A0*, and *S* of the RNBA and the residual error are shown in Figs. 38 and 39, respectively. Through 2.3978×10^4 evolution steps, we can obtain numerical results with a maximum error of 1.921×10^{-4} and *A0* and *S* are 7.9643 and 0.8857, respectively. From the results obtained above, we can conclude that the proposed algorithm has both good stability and high efficiency for solving determinate/indeterminate systems of nonlinear algebraic equations.

IV. CONCLUSIONS

In this paper, the RNBA has been proposed to solve the problems of the non-linear algebraic equations successfully. The fundamental concept starts from a continuous manifold

based on the residual-norm and the ODEs of the specified gradient flow to govern the evolution of the unknown variables. In addition, from a construction of space-time manifold of GPS, we can develop an algorithm with a fictitious time step. Several applications, including of finding the roots of nonlinear algebraic equations in two-, three-variable and elliptic-type PDEs that use a NFDM in high-dimensional space, were tested and illustrated in this paper. Results of the numerical examples demonstrate that the present method can work more effectively and accurately than conventional schemes that use a space-time manifold to solve nonlinear algebraic equations. Most importantly, this method does not involve a complicated computation of the inverse of the Jacobian matrix and need not to solve the function $Q(t)$. Therefore, the RNBA is highly efficient in finding the solutions of determinate or indeterminate systems of non-linear algebraic equations and can enhance the accuracy and the convergence speed.

ACKNOWLEDGMENTS

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

REFERENCES

1. Atluri, S. N., *Methods of Computer Modelling in Engineering and Sciences*, Technical Science Press (2002).
2. Atluri, S. N., Liu, H. T., and Han, Z. D., "Meshless local Petro-Galerkin (MLPG) mixed collocation method for elasticity problems," *Computer Modelling in Engineering & Sciences*, Vol. 14, pp. 141-125 (2006).
3. Atluri, S. N. and Shen, S., "The meshless local Petro-Galerkin (MLPG) method: a simple & less-costly alternative to the finite element and boundary element methods," *Computer Modelling in Engineering & Sciences*, Vol. 3, pp. 11-51 (2002).
4. Atluri, S. N. and Zhu, T., "A new meshless local Petrov-Galerkin (MLPG) approach in computational mechanics," *Computational Mechanics*, Vol. 22, pp. 117-127 (1998).
5. Cheng, A. H. D., Golberg, M. A., Kansa, E. J., and Zammit, G., "Exponential convergence and H-c multiquadric collocation method for partial differential equations," *Numerical Methods for Partial Differential Equations*, Vol. 19, pp. 571-594 (2003).
6. Cho, H. A., Golberg, M. A., Muleshkov, A. S., and Li, X., "Trefftz methods for time dependent partial differential equations," *Computers, Materials & Continua*, Vol. 1, pp. 1-38 (2004).
7. Davidenko, D., "On a new method of numerically integrating a system of nonlinear equation," *Doklady Akademii Nauk SSSR*, Vol. 88, pp. 601-604 (1953).
8. Eigel, M., George, E., and Kirkilionis, M., "A mesh-free partition of unity method for diffusion equations on complex domains," *IMA Journal of Numerical Analysis*, Vol. 30, pp. 629-653 (2010).
9. Gorji, M., Ganji, D. D., and Solenimani, S., "New application of He's homotopy perturbation," *International Journal of Nonlinear Sciences and Numerical Simulation*, Vol. 8, pp. 319-328 (2007).
10. He, J. H., "Homotopy perturbation technique," *Computer Methods in Applied Mechanics and Engineering*, Vol. 178, pp. 257-262 (1999).
11. He, J. H., "Homotopy perturbation methods: a new nonlinear analytical technique," *Applied Mathematics and Computation*, Vol. 135, pp. 73-79 (1999).
12. Hirsch, M. and Smale, S., "On algorithms for solving $f(x) = 0$," *Communications on Pure and Applied Mathematics*, Vol. 32, pp. 281-312 (1979).
13. Hu, H. Y. and Chen, J. S., "Radial basis collocation method and quasi-Newton iteration for nonlinear elliptic problems," *Numerical Methods For Partial Differential Equations*, Vol. 24, pp. 991-1017 (2008).
14. Hu, H. Y., Li, Z. C., and Cheng, A. H. D., "Radial basis collocation methods for elliptic boundary value problems," *Computational Mathematics and Applications*, Vol. 50, pp. 289-320 (2005).
15. Jin, B., "A meshless method for the Laplace and biharmonic equations subjected to noisy boundary data," *Computer Modelling in Engineering & Sciences*, Vol. 6, pp. 253-262 (2004).
16. Kuo, C. Y., Yeih, W. C., and Liu, C. S., "Solving non-linear algebraic equations by a scalar Newton-homotopy continuation method," *International Journal of Non-linear Science and Numerical Simulation*, Vol. 11, pp. 435-450 (2010).
17. Kuo, C. Y., Yeih, W. C., Liu, C. S., and Chi, C. C., "Applications of the fictitious time integration method using a new time-like function," *Computer Modelling in Engineering & Sciences*, Vol. 43, pp. 173-190 (2009).
18. Li, Z. C., Lu, T. T., Hu, H. Y., and Cheng, A. H. D., *Trefftz and Collocation Methods*, WIT Press, Southampton, UK (2008).
19. Liao, S. J., "On the homotopy analysis method for nonlinear problems," *Applied Mathematics and Computation*, Vol. 47, pp. 499-513 (2004).
20. Liu, C. S., "Cone of non-linear dynamical system and group preserving schemes," *International Journal of Non-linear Mechanics*, Vol. 36, pp. 1047-1068 (2001).
21. Liu, C. S., "An effectively modified direct Trefftz method for 2D potential problems considering the domain's characteristic length," *Engineering Analysis with Boundary Elements*, Vol. 31, pp. 983-993 (2007).
22. Liu, C. S., "A highly accurate solver for the mixed-boundary potential problem and singular problem in arbitrary plane domain," *Computer Modelling in Engineering & Sciences*, Vol. 20, pp. 111-122 (2007).
23. Liu, C. S., "A fictitious time integration method for two-dimensional quasilinear elliptic boundary value problems," *Computer Modelling in Engineering & Sciences*, Vol. 33, pp. 179-198 (2008).
24. Liu, C. S., "A time-marching algorithm for solving non-linear obstacle problems with the aid of an NCP-function," *Computers, Materials & continua*, Vol. 8, pp. 53-65 (2008).
25. Liu, C. S., "A fictitious time integration method for solving m-point boundary value problems," *Computer Modelling in Engineering & Sciences*, Vol. 39, pp. 125-154 (2009).
26. Liu, C. S., "A fictitious time integration method for the Burgers equation," *Computers, Materials & continua*, Vol. 9, pp. 229-252 (2009).
27. Liu, C. S. and Atluri, S. N., "A fictitious time integration method (FTIM) for solving mixed complementarity problem with applications to non-linear optimization," *Computer Modelling in Engineering and Sciences*, Vol. 34, pp. 155-178 (2008).
28. Liu, C. S. and Atluri, S. N., "A fictitious time integration method for solving the discretized inverse Sturm-Liouville problem, for specified eigenvalues," *Computer Modelling in Engineering & Sciences*, Vol. 36, pp. 261-285 (2008).
29. Liu, C. S. and Atluri, S. N., "A novel time integration method for solving a large system of non-linear algebraic equations," *Computer Modelling in Engineering & Sciences*, Vol. 31, pp. 71-84 (2008).
30. Liu, C. S. and Atluri S. N., "An enhanced fictitious time integration method for non-linear algebraic equations multiple solutions: boundary layer, boundary value and eigenvalue problems," *Computer Modelling in Engineering & Sciences*, Vol. 59, pp. 301-323 (2010).
31. Liu, C. S. and Atluri, S. N., "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," *Computer Modelling in Engineering & Sciences*, Vol. 71, pp. 279-304 (2011).
32. Liu, C. S., Yeih, W. C., Kuo, C. L., and Atluri, S. N., "A scalar homotopy method for solving an over/under-determined system of non-linear algebraic equations," *Computer Modelling in Engineering & Sciences*, Vol. 53, pp. 47-72 (2009).

33. Marinca, V. and Herisanu, N., "Optimal homotopy perturbation method for strongly nonlinear differential equations," *Nonlinear Science Letter A*, Vol. 1, pp. 273-280 (2010).
34. Wu, T. M., "A study of convergence on the Newton-homotopy continuation method," *Applied Mathematics and Computation*, Vol. 168, pp. 1169-1174 (2005).
35. Zhu, T., Zhang, J., and Atluri, S. N., "A meshless numerical method based on the local boundary integral equation (LBIE) to solve linear and nonlinear boundary value problems," *Engineering Analysis with Boundary Elements*, Vol. 23, pp. 375-389 (1999).