 CMES, vol.99, no.2, pp.123-149, 2014

A Double Iteration Process for Solving the Nonlinear Algebraic Equations, Especially for Ill-posed Nonlinear Algebraic Equations

Weichung Yeih[1,2], I-Yao Chan[1], Cheng-Yu Ku[1],Chia-Ming Fan[1] and Pai-Chen Guan[3]

Abstract: In this paper, a novel double iteration process for solving the nonlinear algebraic equations is developed. In this process, the outer iteration controls the evolution path of the unknown vector $\mathbf{x}$ in the selected direction $\mathbf{u}$ which is determined from the inner iteration process. For the inner iteration, the direction of evolution $\mathbf{u}$ is determined by solving a linear algebraic equation: $\mathbf{B^T B u} = \mathbf{B^T F}$ where $\mathbf{B}$ is the Jacobian matrix, $\mathbf{F}$ is the residual vector and the superscript "$\mathbf{T}$" denotes the matrix transpose. For an ill-posed system, this linear algebraic equation is very difficult to solve since the resulting leading coefficient matrix is ill-posed in nature. We adopted the modified Tikhonov's regularization method (MTRM) developed by Liu (Liu, 2012) to solve the ill-posed linear algebraic equation. However, to exactly find the solution of the evolution direction $\mathbf{u}$ may consume too many iteration steps for the inner iteration process, which is definitely not economic. Therefore, the inner iteration process stops while the direction $\mathbf{u}$ makes the value of a_0 being smaller than the selected margin a_c or when the number of inner iteration steps exceeds the maximum tolerance I_{max}. For the outer iteration process, it terminates once the root mean square error for the residual is less than the convergence criterion ε or when the number of inner iteration steps exceeds the maximum tolerance I_{max}. Six numerical examples are given and it is found that the proposed method is very efficient especially for the nonlinear ill-posed systems.

Keywords: double iteration process, ill-posed, the modified Tikhnov's regularization method.

[1] Department of Harbor and River Engineering & Computation and Simulation Center, National Taiwan Ocean University, Keelung, 202 Taiwan.

[2] Corresponding author, Tel: +886-2-24622192-ext6119, e-mail: wcyeih@mail.ntou.edu.tw

[3] Department of Naval Architecture and System Engineering & Computation and Simulation Center, National Taiwan Ocean University, Keelung, 202 Taiwan.

1 Introduction

The engineering or physical problems are sometimes modeled as nonlinear equations. After discretization, a system of nonlinear algebraic equations is then needed to be solved. Unlike the linear algebraic equation system, there exist not many solvers for the nonlinear algebraic equation system. Among these nonlinear equation solvers, they can be categorized into two folds: iteration and evolution dynamics.

For the group of iteration, most well-known methods are Newton's method [Tjalling (1955)] and Landweber iteration method [Landweber (1951)]. The former one is very efficient for solving nonlinear equation however it is not appropriate to adopt this method for the ill-posed system since the inverse of Jacobian matrix is not easy to obtain. The later one is less efficient than the Newton's method while dealing with the well-posed problem but is more stable when dealing with the ill-posed problem. However, the Landweber iteration method cannot deal with severely ill-posed system and most of times the Tikhonov's regularization method [Tikhonov and Arsenin (1977)] is required.

For the group of evolution dynamics, a system of the first order ordinary differential equations of the unknowns is constructed and the trajectory of the unknown will approach to the fixed point of this ODE system which is the solution of the original algebraic equation system [Ramm (2007)]. The homotopy method [Billups (2002)], the scalar homotopy method [Liu, Yeih, Kuo and Atluri (2009)], the fictitious time integration method [Ku, Yeih, Liu and Chi (2009)] and so on can be categorized in this group.

No matter the iteration method or the evolution dynamic method is adopted, the unknown vector changes according to some known direction for most methods. For example, the Newton's method uses the direction of $\mathbf{B}^{-1}\mathbf{F}$ where the superscript '-1' denotes the inverse of a matrix, Landweber iteration method and the exponentially convergent scalar homotopy algorithm (ECSHA) [Chan, Fan and Yeih (2011)] use the direction of $\mathbf{B}^{T}\mathbf{F}$. The fictitious time integration method (FTIM) and the dynamic Jacobian inverse free method (DJIFM) [Ku, Yeih and Liu (2011)] adopt the direction of $\mathbf{F}$. Liu [Liu and Atluri (2011a)] proposed to use two directions at the same time and he found the optimal combination of these two directions. [Yeih, Ku, Liu and Chan (2013)] extended this idea and answered the question for finding the optimal combination for multiple directions. All these methods adopt one or a combination of multiple known directions, however, generally speaking adopting a known direction or a combination of multiple known direction cannot guarantee it will be the best one for all problems. Later in the article, it will be found that theoretically the optimal direction will be the direction of $\mathbf{B}^{-1}\mathbf{F}$ if the

inverse of the Jacobian matrix does exist. However, for the ill-posed problem or some special cases the inverse of the Jacobian matrix does not exist or cannot be found in the numerical sense the optimal searching direction used in the iteration process or evolution dynamics method is then not proposed to authors' best knowledge. To overcome the ill-posed nature, the abovementioned alternatives such as the Landweber iteration, the fictitious time integration method and so on adopt other directions than the direction of $\mathbf{B}^{-1}\mathbf{F}$ such that the numerical process will be stabler. The problem is these methods show slow convergence rate which makes solving the nonlinear ill-posed problem become not economic in numerical sense. Especially when the large scale problem is encountered, computation effort to deal with the ill-posed nature then becomes awful and not acceptable for engineers.

The method proposed here combine two recent developed methods: the residual norm based algorithm and the modified Tikhonov's regularization method. The followings give a brief review of these two methods. Recently, the residual norm based algorithm (RNBA) has been proposed to deal with the nonlinear algebraic equation system. [Liu and Atluri (2012); Liu and Atluri (2011b)] The RNBA basically is a type of the scalar homotopy method [Liu, Yeih, Kuo and Atluri (2009)] where the trajectory of the unknown vector is required to lie on the space-time manifold. The RNBA constructs an iteration process from the evolution dynamics when the evolution direction $\mathbf{u}$ is selected. Later, Liu (2013) reported that the value of the relaxation parameter in RNBA has an optimal value. The modified Tikhonov's regularization method (MTRM) [Liu (2012)] proposed an iteration to solve the solution for an ill-posed linear system. In the same paper, Liu also proposed a generalized Tikhnov's regularization method (GTRM) to solve the ill-posed linear system. The MTRM is very similar to conventional Tikhonov's regularization method which adds a regularization parameter in the diagonal line of the leading coefficient matrix while the GTRM adds regularization parameters in the diagonal line and the determination for these regularization parameters depend on the equilibrate matrix concept.

In this article, we develop a novel double iteration process to deal with the nonlinear algebraic equation systems. For outer iteration process, the evolution path of the unknown vector follows the searching direction determined from the inner iteration process and the process requires the path falls on the space-time manifold such that the convergence rate can be guaranteed. To determine the searching direction, we solve a linear algebraic equation system: $\mathbf{B^T B u} = \mathbf{B^T F}$. For a well-posed problem, it can be easily proved that the direction $\mathbf{u}$ for the above problem is $\mathbf{B}^{-1}\mathbf{F}$**.** However, for the ill-posed problem the above linear system cannot be solved due to the ill-posed nature. We adopt the modified Tikhonov's regularization method (MTRM) [Liu (2012)] to iteratively approach the solution for the above linear sys-

tem. However, for ill-posed problems to really find the solution of **u** may require too many iteration steps for the modified Tikhonov's regularization method which makes the whole numerical process not economic at all. Therefore, we propose that the inner iteration process should stop while the direction **u** already makes the value of a_0 being smaller than the prescribed critical value a_c(it should be smaller than 4 to guarantee the path falls on the manifold) or while the number of the iteration steps for the current inner iteration process exceeds the prescribed maximum tolerance value says I_{max}. The former criterion loosen the problem for solving $\mathbf{B^T B u} = \mathbf{B^T F}$ exactly (for which the value of a_0 should be one exactly) by finding an approximated direction such that $a_0 < a_c$ and the path is still guaranteed on the manifold. The later criterion avoids consuming too many iteration steps for the inner iteration process. It means when one needs to consume too many iteration steps for the inner iteration process the numerical process becomes not economic and should be stopped immediately. Six numerical examples are illustrated to show the validity and efficiency for the proposed method. It is found that the proposed method show excellent efficiency especially for the ill-posed nonlinear problem. Aside this section, other sections will be arranged as the followings. In the second section, the mathematical backgrounds used in this article will be introduced. In the third section, six numerical examples including the Brown's problem, the stagnant point problem, the nonlinear Fredholm integral equation of the first kind, the problem of determining the unknown boundary for the biharmonic equation, the problem of finding the thermal conductivity and the temperature for the heat equilibrium equation by prescribing the surface temperature and the surface distribution of thermal conductivity in advance and the inverse Cauchy problem for the nonlinear semi-linear partial differential equation. In the final section, the conclusions will be drawn based on the results from this article.

2 Mathematical backgrounds

2.1 Residual Norm Based Algorithm (RNBA)

The following derivation can be found in many related articles such as [Liu and Atluri (2012); Liu and Atluri (2011b)]. Let us begin with a nonlinear algebraic system written as:

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

where **F** denotes the residual vector and **x** denotes the unknown vector. To solve this nonlinear algebraic equation system, we formulate an equivalent scalar equation written as

$$\|\mathbf{F}(\mathbf{x})\|^2 = 0. \tag{2}$$

It is obvious that solving equation (1) is equivalent to solving equation (2) and vice versa. Now let us construct a space-time manifold as:

$$h(\mathbf{x},t) = \frac{1}{2}\|\mathbf{F}(\mathbf{x})\|^2 - \frac{1}{2}\frac{1}{Q(t)}\|\mathbf{F}(\mathbf{x}_0)\|^2 = 0 \tag{3}$$

where $\mathbf{x}_0$ is the initial guess and $Q(t)$ satisfies that $Q(t) > 0$, $Q(0) = 1$, and it is a monotonically increasing function of t with $Q(\infty) = \infty$.

In order to keep the trajectory of the solution $\mathbf{x}$ on the manifold, the following consistency equation should be satisfied:

$$\frac{Dh}{Dt} = \frac{\partial h}{\partial t} + \nabla h \cdot \frac{d\mathbf{x}}{dt} = 0, \tag{4}$$

where ∇denotes the gradient operator. Since equation (4) is a scalar equation, it is not possible to determine the evolution of the unknown vector (i.e., $\frac{d\mathbf{x}}{dt}$) uniquely. Let us assume that the evolution of the unknown vector is in the direction of $\mathbf{u}$ and we have:

$$\dot{\mathbf{x}} = \frac{d\mathbf{x}}{dt} = \lambda\mathbf{u}, \tag{5}$$

where λ is the proportional constant. After some manipulations, we can obtain the evolution equation of $\mathbf{x}$ as

$$\dot{\mathbf{x}} = -\frac{\dot{Q}(t)}{2Q(t)}\frac{\|\mathbf{F}(\mathbf{x})\|^2}{\mathbf{F}^{\mathrm{T}}(\mathbf{x})\mathbf{v}}\mathbf{u} \tag{6}$$

where **v=Bu**. Now let us consider the evolution of the residual vector as:

$$\dot{\mathbf{F}}(\mathbf{x}(t)) = \mathbf{B}\dot{\mathbf{x}}. \tag{7}$$

Substituting equation (6) into equation (7), we then obtain:

$$\dot{\mathbf{F}}(\mathbf{x}(t)) = \frac{-\dot{Q}(t)}{2Q(t)}\frac{\|\mathbf{F}(\mathbf{x})\|^2}{\mathbf{F}^{\mathrm{T}}(\mathrm{x})\mathbf{v}}\mathbf{v} \tag{8}$$

Using the forward Euler scheme, we can discretize equation (8) as:

$$\mathbf{F}(\mathbf{x}(t+\Delta t)) = \mathbf{F}(\mathbf{x}(t)) - \Delta t\frac{\dot{Q}(t)}{2Q(t)}\frac{\|\mathbf{F}(\mathbf{x})\|^2}{\mathbf{F}^T(\mathbf{x})\mathbf{v}}\mathbf{v}. \tag{9}$$

where Δt is the time increment. Using equation (9), one can have $\|\mathbf{F}(\mathbf{x}(t))\|^2 = \frac{\|\mathbf{F}(\mathbf{x}_0)\|^2}{Q(t)}$ and $\|\mathbf{F}(\mathbf{x}(t+\Delta t))\|^2 = \frac{\|\mathbf{F}(\mathbf{x}_0)\|^2}{Q(t+\Delta t)}$.

Taking square norm of the above expressions and using $\beta := \Delta t \frac{\dot{Q}(t)}{2Q(t)}$ and equation (9), one can derive an algebraic equation as:

$$\frac{\|\mathbf{F}(\mathbf{x}_0)\|^2}{Q(t+\Delta t)} = \frac{\|\mathbf{F}(\mathbf{x}_0)\|^2}{Q(t)} - 2\beta\frac{\|\mathbf{F}(\mathbf{x}_0)\|^2}{Q(t)} + \beta^2\frac{\|\mathbf{F}(\mathbf{x}_0)\|^2}{Q(t)}\frac{\|\mathbf{F}(\mathbf{x})\|^2}{(\mathbf{F}^T(\mathbf{x})\mathbf{v})^2}\|\mathbf{v}\|^2.$$

Consequently, an algebraic equation for β is obtained as:

$$a_0\beta^2 - 2\beta + 1 - \frac{Q(t)}{Q(t+\Delta t)} = 0, \tag{10}$$

where $a_0 = \frac{\|\mathbf{F}(\mathbf{x})\|^2\|\mathbf{v}\|^2}{(\mathbf{F}^T(\mathbf{x})\mathbf{v})^2} = \left\{\frac{\|\mathbf{F}(\mathbf{x})\|\|\mathbf{v}\|}{(\mathbf{F}^T(\mathbf{x})\mathbf{v})}\right\}^2 = \left(\frac{1}{\cos\theta}\right)^2$ in which θ denotes the angle between the residual vector $\mathbf{F}$ and the vector $\mathbf{v}$.

From the Cauchy-Schwarz inequality, it can be easily verified that $a_0 \geq 1$. Now let us define $s := \frac{Q(t)}{Q(t+\Delta t)} = \frac{\|\mathbf{F}(\mathbf{x}(t+\Delta t))\|^2}{\|\mathbf{F}(\mathbf{x}(t))\|^2}$, it can be found that this ratio s is the ratio between the square norm of the residual vector in the next state and the square norm of the residual vector in the current state. It is for sure that we hope $s \leq 1$, such that for each state the norm of the residual vector decreases. The equation (10) now can be rewritten as $a_0\beta^2 - 2\beta + 1 - s = 0$ and we can obtain real-valued $\beta = \frac{1-\sqrt{1-(1-s)a_0}}{a_0}$ if 1-(1-s)$a_0 \geq 0$. For simplicity, we let $1-(1-s)a_0 = r^2$ (r is a relaxation parameter which will be explained later) and use the definition of a_0, one can obtain:

$$s = 1 - \frac{\left(1-r^2\right)\left(\mathbf{F}^T(\mathbf{x})\mathbf{v}\right)^2}{\|\mathbf{F}(\mathbf{x})\|^2\|\mathbf{v}\|^2}. \tag{11}$$

Now let us use the forward Euler scheme on equation (6), we can obtain the following equation

$$\mathbf{x}(t+\Delta t) = \mathbf{x}(t) - (1-r)\frac{(\mathbf{F}^T(\mathbf{x})\mathbf{v})}{\|\mathbf{v}\|^2}\mathbf{u}. \tag{12}$$

For a selected value of r, we can rewrite equation (12) as an iteration formula [Liu and Atluri (2011b)]:

$$\mathbf{x}_{k+1} = \mathbf{x}_k - (1-r)\frac{\mathbf{F}^T(\mathbf{x}_k)\mathbf{v}_k}{\|\mathbf{v}_k\|^2}\mathbf{u}_k. \tag{13}$$

In the above equation, the relaxation parameter is used to make the iteration stabler. In a recent published paper, Liu (2013) further found the value of r needs to satisfy the following relationship to guarantee the trajectory of $\mathbf{x}$ remains on the manifold:

$$r = \left\| 1 - \frac{a_0}{2} \right\|. \tag{14}$$

And Liu (2013), Ku and Yeih (2012) all reported that the value of a_0 is between 1 and 4 if we hope the trajectory of $\mathbf{x}$ remain on the manifold. From the definition of a_0, we know that the value of a_0 relates to the vector $\mathbf{F}$ and $\mathbf{v}$ (or equivalently $\mathbf{u}$). The problem now is how to find a vector $\mathbf{v}$ (or equivalent $\mathbf{u}$) such that a_0 is between 1 and 4. If such a direction is found, we then select the value of r as $r = \left\| 1 - \frac{a_0}{2} \right\|$ in equation (14). To find an appropriate direction $\mathbf{u}$ then becomes the key. Theoretically speaking, if a_0=1 then the residual norm decreases in the fast manner. This means that the best direction $\mathbf{u}$ will satisfy that $\mathbf{Bu} - \mathbf{F} = \mathbf{0}$ (or $\mathbf{u} = \mathbf{B}^{-1}\mathbf{F}$ if the inverse of the Jacobian matrix exists), it means that the Newton's iteration method is the best iteration. However, for ill-posed systems to seek the inverse of the Jacobian matrix sometimes is impossible due to its numerical instability. Therefore, one requires an algorithm to find an appropriate $\mathbf{u}$ for the nonlinear ill-posed systems. To achieve this, we first give a brief review of the modified Tikhonov's regularization method (MTRM) as the following subsection describes.

3 Modified Tikhonov's Regularization Method (MTRM)

The regularization technique is well-known for dealing with an ill-posed system. There exist many literatures mentioning the regularization technique. For readers' convenience, the following references can provide a conceptual understanding for the regularization technique. [Lin, Chen and Wang (2011); Wang, Chen and Ling (2012); Fu, Chen and Zhang (2012)]

The details of the following descriptions can be found in [Liu (2012)]. Considering the following linear algebraic system as:

$$\mathbf{Bu} = \mathbf{F} \tag{15}$$

and we use the following preconditioner written as:

$$\mathbf{P}_1 = \mathbf{B}^T + \bar{\alpha}\mathbf{B}^+, \tag{16}$$

where $\mathbf{B}^+$ is the pseudo-inverse with $\mathbf{B}^+\mathbf{B} = \mathbf{I}$.

and apply this preconditioner to equation (15) then we will obtain

$$\left(\mathbf{B}^T\mathbf{B} + \bar{\alpha}\mathbf{I}\right)\mathbf{u} = \mathbf{B}^T\mathbf{F} + \bar{\alpha}\mathbf{u}. \tag{17}$$

It is quite interesting to find that the regularized equation in equation (17) is very similar to that of the conventional Tikhonv's regularization method. However, in equation (17) the regularization parameter appears in the both sides of equation while for the conventional Tikhonov's regularization method it appears only in the left-hand side.

Liu (2012) proposed that one can use equation (17) to formulate an iteration process as:

$$\left(\mathbf{B}^T\mathbf{B}+\bar{\alpha}\mathbf{I}\right)\mathbf{u}_{p+1}=\mathbf{B}^T\mathbf{F}+\bar{\alpha}\mathbf{u}_p \tag{18}$$

The convergence criterion of the iteration process for equation (18) can be set as: $\|\mathbf{u}_{p+1}-\mathbf{u}_p\|\leq\varsigma$ where ς is a preselected small tolerance. Liu also provided a theoretical proof of the convergence as the following theorem states:

[Theorem 1] For Eq. (18) with $\bar{\alpha}>0$ the iterative sequence $\mathbf{u}_p$converges to the true solution $\mathbf{u}_{true}$ monotonically.

Although the convergence of the sequence is guaranteed, in computation reality to reach the final numerical convergence it may takes too many steps such that it becomes not economic at all. It means that if one tries to find the solution of an ill-posed linear system, a lot of computation effort will be paid for the iteration process (equation (18)) and sometimes it makes this iteration not economic at all.

This algorithm needs to be further examine while it is used to solve the best direction **u** such that **Bu-F=0** since for each step in the iteration process stated in equation (13) for solving the nonlinear problem this linear algebraic equation **Bu-F=0** needs to be done if one tries to find the optimal direction. However, to find the solution of this linear problem may cost too many iteration steps for iteration process equation (18). Remember that we are not really interested in finding the best direction we only want to find an appropriate **u** such that a_0 is between 1 and 4. Therefore, we can check this criterion for each step of the inner iteration (equation (18)) and terminate the inner iteration as the value of a_0 is less than a prescribed critical value a_c. Of course, it may take too many steps to let a_0 being less than this prescribed critical value a_c. We say that if the number of iteration steps for the inner loop exceeds a preselected maximum number I_{max}, we then stop the inner loop as well as the outer loop. It means that to find an appropriate direction of evolution using the proposed algorithm already becomes not economic and the whole process should be terminated.

4 Double Iteration Process (DIP)

Based on the abovementioned theoretical backgrounds, we proposed a double iteration process as the followings.

Double Iteration Process (DIP):

Give initial guess $\mathbf{x_0}$

Give prescribed parameters ε, $\overline{\alpha}$.

Outer Iteration:

For k=0,1,2,… Repeat

Calculate the residual vector $\mathbf{F}_k(\mathbf{x}_k)$ and the Jacobian matrix $\mathbf{B}_k(\mathbf{x}_k)$

Inner Iteration:

Give the initial guess of $\mathbf{u}$ as $\mathbf{u}_0 = \dfrac{\mathbf{B}_k{}^T\mathbf{F}_k}{\left\|\mathbf{B}_k{}^T\mathbf{F}_k\right\|}$,

For p=1,2,…, I_{max}

Solve $\mathbf{u}_{p+1}$ by $\left(\mathbf{B}^T\mathbf{B}+\overline{\alpha}\mathbf{I}\right)\mathbf{u}_{p+1} = \mathbf{B}^T\mathbf{F}+\overline{\alpha}\mathbf{u}_p$

Construct $\mathbf{v}_{p+1} = \mathbf{B}_k\mathbf{u}_{p+1}$

$$\left(a_0\right)_{p+1} = \frac{\left\|\mathbf{F}_k\left(\mathbf{x}\right)\right\|^2\left\|\mathbf{v}_{p+1}\right\|^2}{\left(\mathbf{F}_k{}^T\left(\mathbf{x}\right)\mathbf{v}_p\right)^2}$$

(a) If $\left(a_0\right)_{p+1} \le a_c$, then $\left(a_0\right)_k = \left(a_0\right)_{p+1}$, $\mathbf{u}_k = \mathbf{u}_{p+1}$, $\mathbf{v}_k = \mathbf{v}_{p+1}$ and one terminates the inner iteration, otherwise continue.

(b) If p=I_{max}, terminates the whole process.

End of Inner Iteration

Calculate $r_k = \left\|1-\dfrac{\left(a_0\right)_k}{2}\right\|$

$$\mathbf{x}_{k+1} = \mathbf{x}_k - (1-r)\frac{\mathbf{F}^T\left(\mathbf{x}_k\right)\mathbf{v}_k}{\left\|\mathbf{v}_k\right\|^2}\mathbf{u}_k$$

If RMSE$\le\varepsilon$ or (b) is true then the outer iteration process stops; otherwise continue.

End of Outer Iteration Process.

It is worth mentioned here that the initial guess in the inner iteration uses the descent direction. Actually, one can select other alternatives such as $\mathbf{u}_0$=**0**. How the initial guess for the inner iteration process influences the accuracy and efficiency of DIP leaves as an open question and in this article we use the initial guess as mentioned above. The DIP can be summarized in the flow chart in Fig. 1.

From the abovementioned double iteration process, we can find that the proposed method does not really try to solve the linear algebraic equation **Bu-F=0** since it

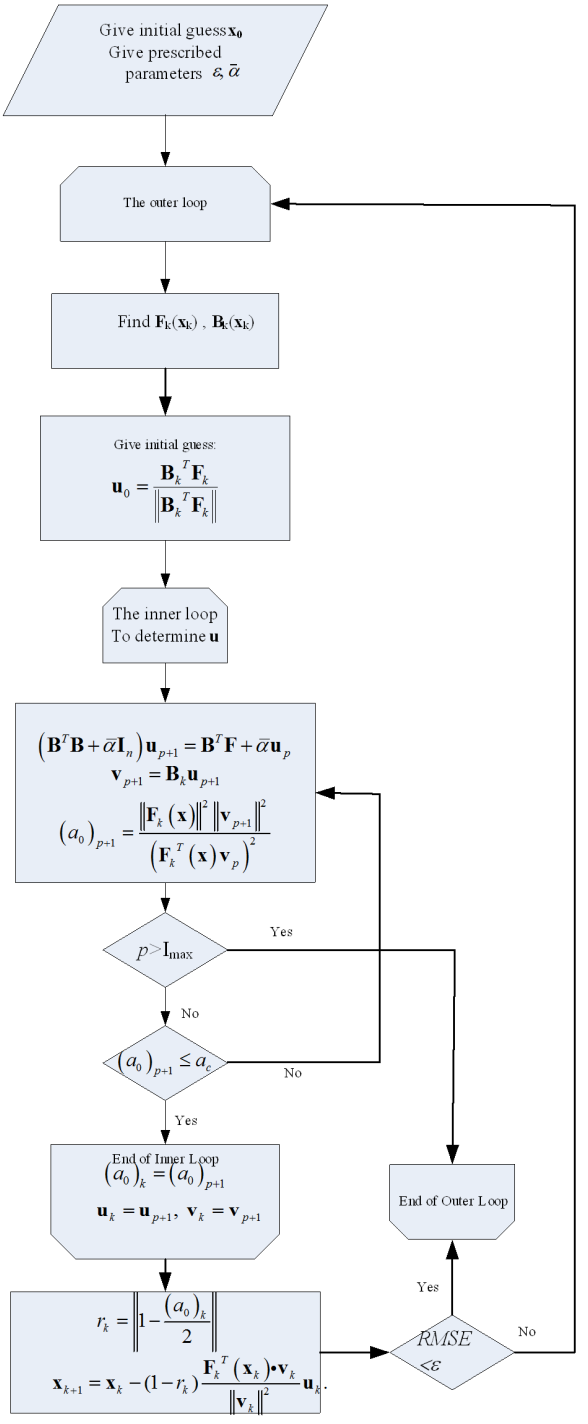


Figure 1: The flow chart of DIP.

is expected that for the ill-posed system it may take too many iteration steps to accomplish this for the inner iteration and it definitely costs too much. In order to avoid using much computation effort, we say once the value of a_0 is less than the prescribed value a_c we claim that the appropriate direction has been found already. Of course we need to remind ourselves that while a_c approaches to one the inner iteration takes more and more steps. And we expect that to find an appropriate direction may still require unreasonably many steps so we say that once the iteration steps exceed that maximum value I_{max} we can stop the whole process since it becomes not economic for further searching. The appropriate values of a_c , $\bar{\alpha}$ and I_{max} influence the convergence speed a lot and how to choose them leaves as an open question.

According to our numerical experiences, the value of a_c is suggested to in the range from 2.5 to 4. Once the value of a_c is less than 2.5, to seek the appropriate direction in the inner loop then consumes too many iteration steps. The value of I_{max} actually depends on the selection of a_c. If the value of a_c is between 2.5 and 4, the value of I_{max} is suggested to be in the range of 30,000 to 80,000 according to our numerical experiences. The selection of ε depends on the system we want to solve. If the system is a well-posed system, the value of ε can be very small such as 10^{-7}. However, if the system is an ill-posed system the value of ε should not be very big and usually 10^{-3} or 10^{-4} is appropriate. It should be mentioned here that actually for an ill-posed system the value of ε can be set as a small value for DIP since this tight convergence criterion cannot be reached and the whole DIP will be terminated due to the number of the inner iteration steps exceeds the maximum value I_{max}. It means the selection of ε is not critical at all. The value of $\bar{\alpha}$ needs to be larger than the smallest eigenvalue of $\mathbf{B}^T\mathbf{B}$ which varies step by step. In calculation reality, a big enough value is selected. However, if $\bar{\alpha}$ is too big the iteration process for eq.(18) becomes slow.

5 Numerical Examples

[Example 1] In this example, we consider an almost linear problem [Brown (1973)]:

$$F_i = x_i + \sum_{j=1}^{j=n} x_j - (n+1) = 0,\ i = 1,\cdots,n-1 \quad \text{and} \quad F_n = \prod_{j=1}^{j=n} x_j - 1 = 0 \tag{19}$$

with a closed-form solution as $x_i = 1$, for $i = 1,\cdots,n$. We select n=100 and the initial guess of the unknown is set as $\mathbf{x}_0 = \begin{bmatrix} 0 & \cdots & 0 \end{bmatrix}^T$. It is easy to find that for the initial guess point the associate Jacobian matrix is singular, i.e., the inverse of the Jacobian does not exist at all. The parameters used for the double iteration process are $\bar{\alpha}$=0.1, a_c=2.5, I_{max}=20000 and ε=10^{-7}. From Fig. 2, we can find that

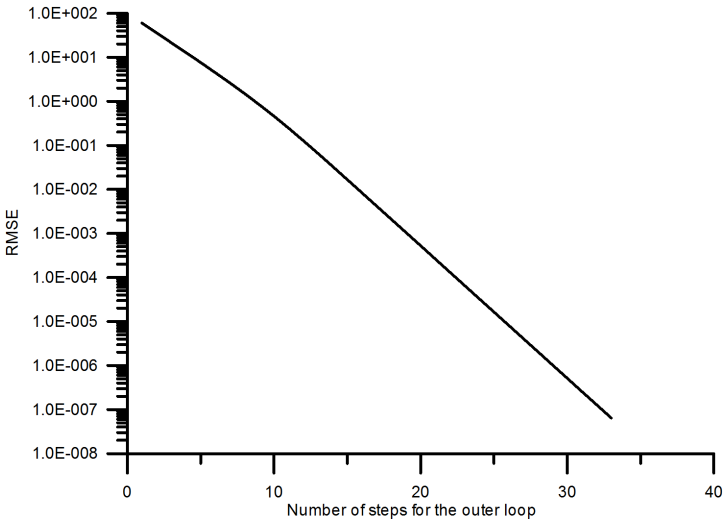


Figure 2: The RMSE versus the number of steps for the outer loop in example 1.

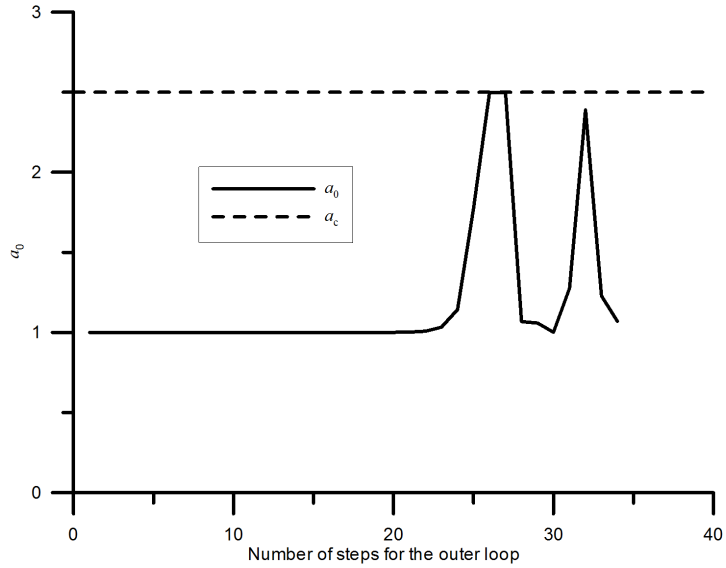


Figure 3: The value of $a_0$never exceeds a_c.

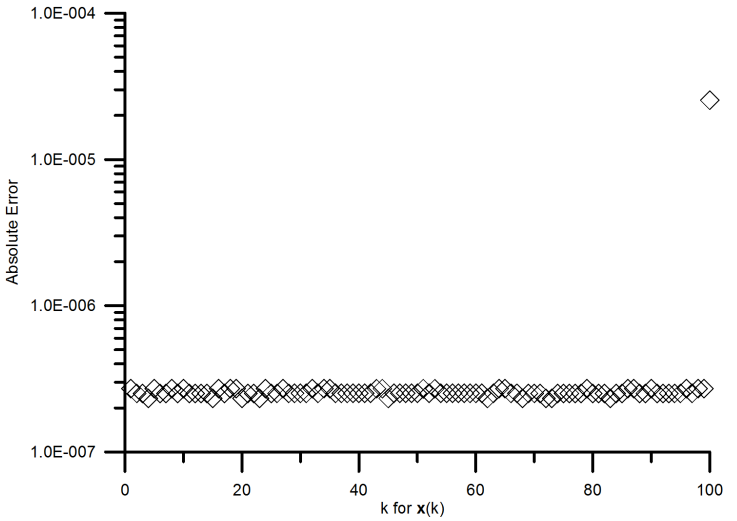


Figure 4: The absolute error for the k-th component in the solution vector **x**.

after 34 steps for the outer loop the RMSE reaches the convergence criterion. In addition, we can find from Fig. 3 that for each step for the outer loop the value of a_0 never exceeds the critical value a_c =2.5. From Fig. 4, one can find that the numerical solution is very close to the exact solution and the maximum absolute error occurs for the last component of the solution vector **x**, i.e., **x**(100). Nevertheless, the maximum absolute error is less than 10^{-4}. It is worth mentioned here if one tries to solve this problem by the conventional Newton method using the initial guess given above, he will not obtain correct solution since the Jacobian matrix becomes singular.

[Example 2] In this problem, we consider the following simple system:

$$\begin{aligned} F_1(x_1,x_2) &= x_1^2 + x_2^2 - 2 = 0, \\ F_2(x_1,x_2) &= e^{(x_1-1)} + x_2^2 - 2 = 0, \end{aligned} \tag{20}$$

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

This is an interesting example because the iteration for Newton's method fails when the initial guess is selected as (3,5) as shown in Fig. 5. As the trajectory approaches to $x_1 = 3.5192$ during the iteration, it happens $x_2 \approx 0.0$. It then is found that the Jacobian matrix now is nearly singular. This leads the trajectory of (x_1, x_2) oscillates at the axis for $x_1 = 3.5192$ as shown in Fig. 5. It means that the conventional Newton method fails for this case. We now use the double iteration process to solve

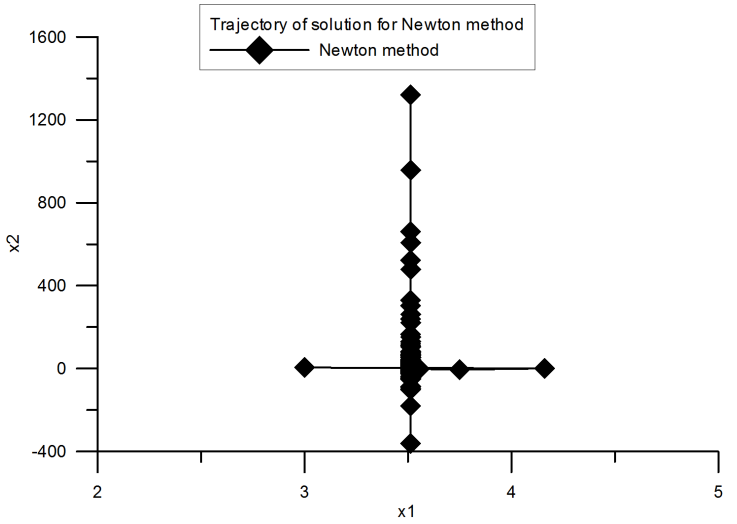


Figure 5: Trajectory for the solution using the Newton method.

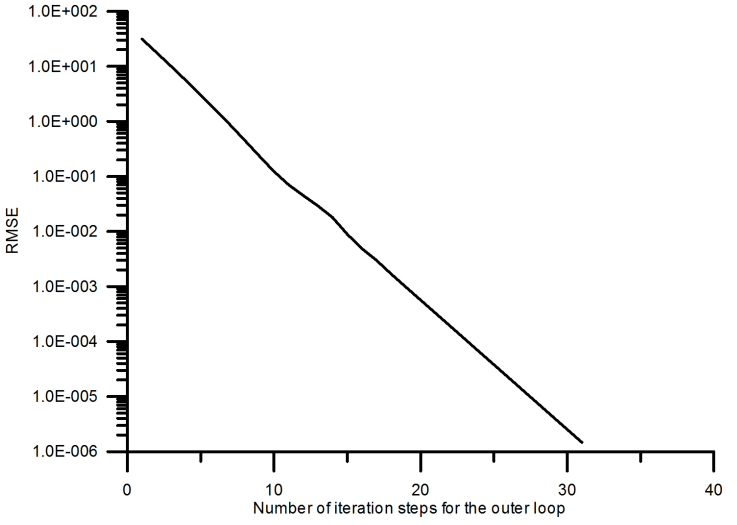


Figure 6: RMSE versus the number of iteration steps for the outer loop in example 2.

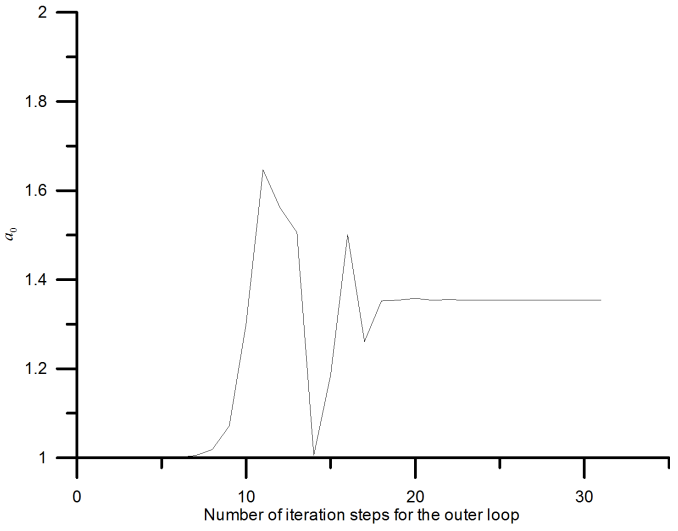


Figure 7: The value of a_0 never exceeds the critical value a_c=2.0.

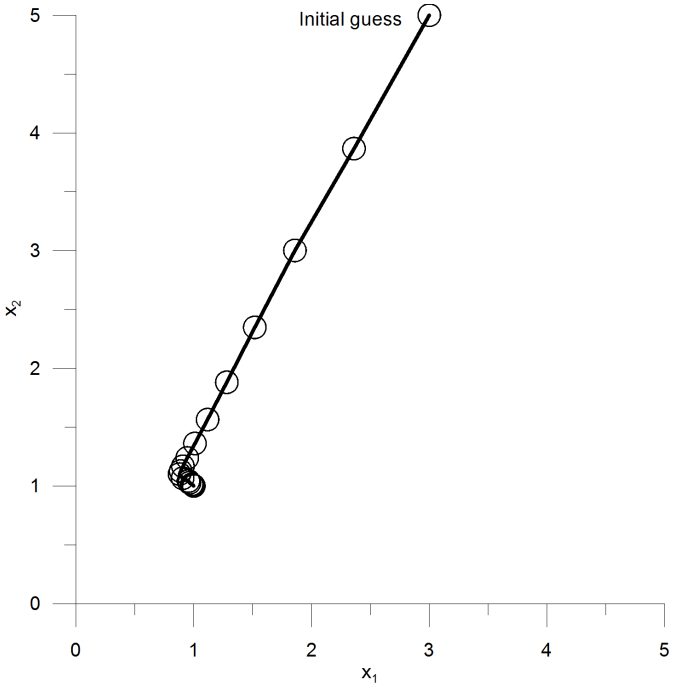


Figure 8: The trajectory of the solution for the double iteration method.

this problem with the initial guess is set as $(x_1, x_2) = (3, 5)$. The parameters used for the double iteration process are $\bar{\alpha}$=10, a_c =2.0, I_{max}=30000 and ε=10^{-6}. It is found from Fig. 6 that the process terminates after 31 steps for the outer loop. We check the plot of a_0 as shown in Fig. 7 and we find out that a_0 never exceeds the critical value a_c =2.0 which once more shows that our method really can guarantee the trajectory of the solution vector falls on the manifold. The trajectory of the solution is shown in Fig. 8.

[Example 3] A classical example of an ill-posed problem is the nonlinear Fredholm integral equation of the first kind which is well known as a nonlinear ill-posed problem. The problem we consider is written as:

$$\int_0^1 x(s)\,x(t)\,dt = \mathrm{A}\cos\left(\bar{\beta}s\right),\ \mathrm{A} > 0 \tag{21}$$

where A and $\bar{\beta}$ are constants. We let A=1 and $\bar{\beta} = 3$ in the followings. We give data for $\mathrm{A}\cos(\bar{\beta}s)$ in the region $s \in [0,1]$ by equally dividing the region into 200 segments, that means totally 201 data points are used. The data has disturbed by maximum 5% relative error. Therefore, we also use these 201 points as the integration quadrature points and the trapezoidal rule is used for integration.

Two exact solutions exist: $x(s) = \pm\sqrt{\frac{\mathrm{A}\beta}{\sin\beta}}\cos(\beta s) = \pm\sqrt{\frac{3}{\sin 3}}\cos(3s)$ [Polyanin and Manzhirov (2007)]. The initial guess are given as $x(t) = 10.0$ for $t \in [0,1]$ and the following parameters are used: $\bar{\alpha}$=0.1, a_c =2.5, I_{max}=30000 and ε=10^{-3}. The reason why the value of ε is not very small in comparison with the previous two examples is that this problem is ill-posed in nature and therefore the convergence criterion can be larger. From Fig. 9, one can find that after 30 steps for the outer loop the solution converges to the requirement. It can be found from Fig. 10 that the numerical solution is acceptable for an ill-posed problem with maximum 5% relative absolute random error in data. The relative absolute random error percentage (RAREP) for a given data x_{given} (from numerical calculation or prescribed value) is defined as:

$$\mathrm{RAREP} = \frac{\|x_{given} - x_{true}\|}{\|x_{true}\|} \times 100\%$$

The proposed method shows excellent noise resistance for the ill-posed nonlinear problem.

[Example 4] The following problem appeared in [Chan and Fan (2013)]. Here we consider a boundary detection problem with the governing equation is the biharmonic equation. The boundary enclosing the computational domain is defined by

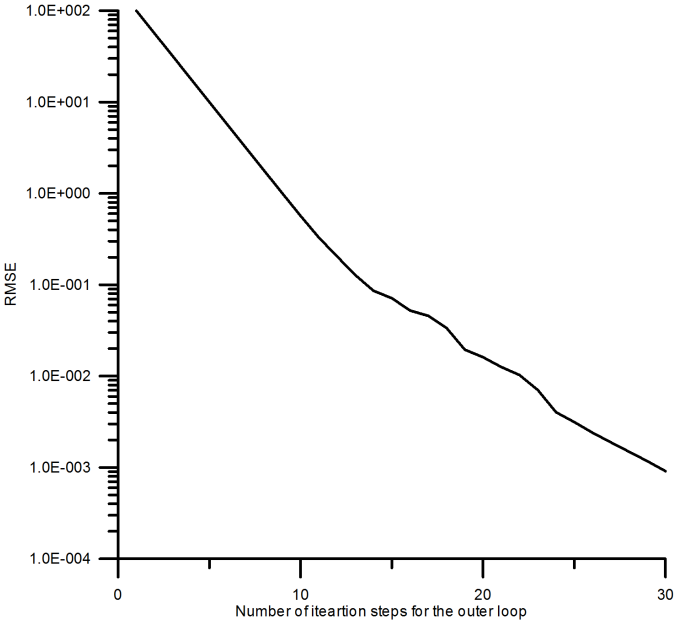


Figure 9: RMSE versus number of iteration steps for the outer loop in example 3.

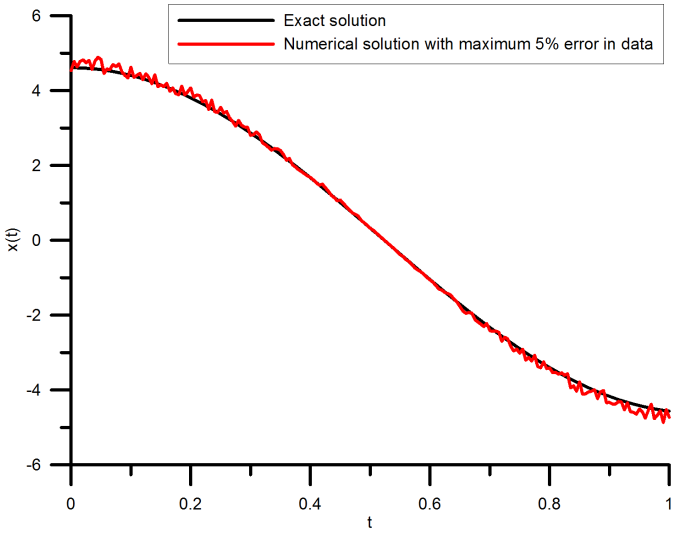


Figure 10: The solution of a Fredholm integral equation of the first kind.

the parametric equation:

$$\Gamma_0 = \{(x,y)\,|x = \rho\cos\theta, y = \rho\sin\theta, 0 \le \theta \le 2\pi\} \tag{22}$$

where

$$\rho(\theta) = 1 + \cos^2(4\theta). \tag{23}$$

For $0 \le \theta \le \pi$, we prescribe four boundary data as

$$u(x,y) = p_1(x,y) \tag{24}$$

$$\frac{\partial u(x,y)}{\partial n} = p_2(x,y) \tag{25}$$

$$w(x,y) = \nabla^2 u(x,y) = p_3(x,y) \tag{26}$$

$$\frac{\partial w(x,y)}{\partial n} = p_4(x,y). \tag{27}$$

For $\pi < \theta \le 2\pi$, only the Dirichlet boundary data is given but the geometry or the lower part is missing. We assume that the designed field property is written as:

$$u(x,y) = e^x \cos y + x^3 - y^3 + 2. \tag{28}$$

To recover the missing boundary and solve the field quantity at the same time makes this problem becomes a nonlinear ill-posed inverse problem. To solve this problem, we adopt the same discretization method used in [Chan and Fan (2013)], that means the modified Trefftz collocation method is used. For the known boundary, totally 60 points are arranged and 40 points are used for the unknown boundary and the initial guess of the missing boundary is a half circle. The characteristic length used in this problem is 3.0 and the order for the basis is 24. That means totally we have 98 Trefftz basis functions, for more details please refer to [Chan and Fan (2013)]. The parameters used for double iteration process are: $\bar{\alpha}$=10, a_c =2.5, I_{max}=50000 and ε=10^{-3}. It can be found from Fig. 11 that the RMSE never reaches the requirement and the whole process terminates for the number of iteration steps for the inner loop is equal to 22. The reported CPU time is 106.69 sec using the Pentium ® dual core CPU E5200 at 2.5GHz. The accumulated steps for the inner loop are 70659 steps. We further examine the plot of a_0 as shown in Fig. 12, the value of a_0 exceeds the critical value a_c and it seems that further seeking an appropriate vector **u** becomes numerically uneconomic. However, we can find in Fig. 13 that the recovering shape for the missing boundary is already acceptable. The proposed method can automatically stop while further reducing the norm of residual vector becomes difficult and stops at that time still yield acceptable result.

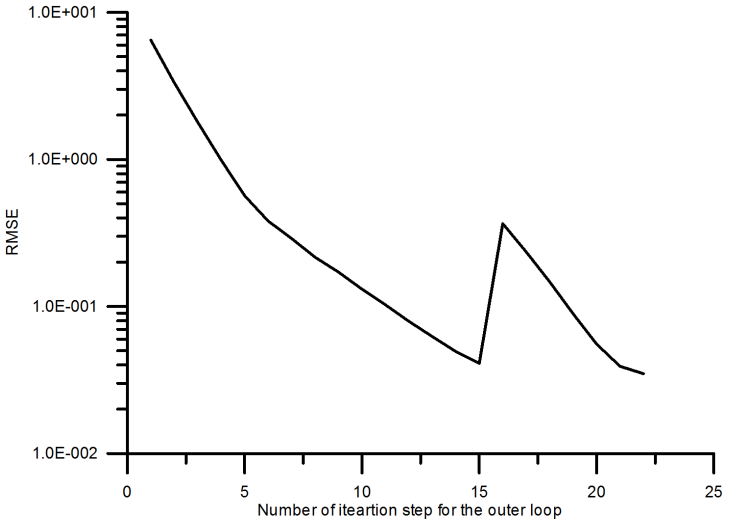


Figure 11: RMSE versus the number of iteration step for the outer loop in example 4.

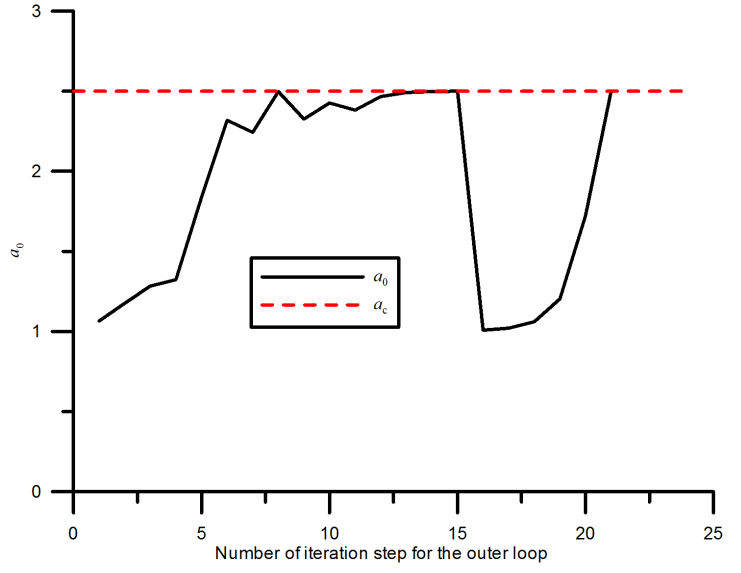


Figure 12: The value of a_0 exceeds a_c in example 4.

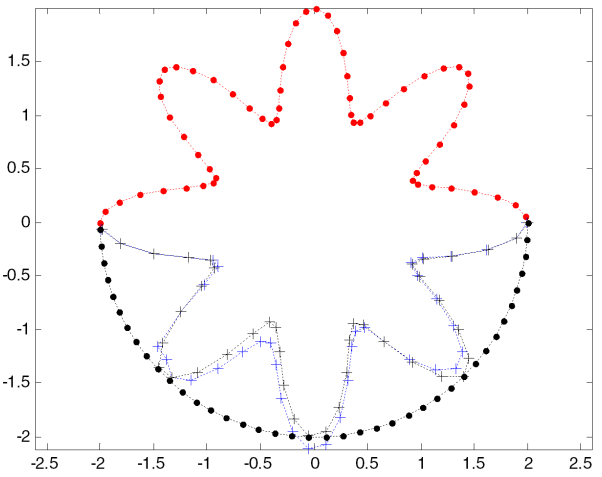


Figure 13: The recovery of missing boundary. (-.-: known boundary; -.- : initial guess for the unknown boundary; - +-: exact unknown boundary; -+-: recovery of missing boundary using data with maximum 1% relative error)

[Example 5] In the following, an inverse problem is given as :

$\nabla \cdot (\sigma(\mathbf{x}) \cdot \nabla u(\mathbf{x})) = 0$ for $\mathbf{x} \in \Omega$

where Ω is the interested domain.

The field quantity u and the conductivity σ is both unknown. The boundary values of the field quantity and conductivity are given as boundary conditions:

$$u(\mathbf{x}) = f_1(\mathbf{x}) \text{ for } \mathbf{x} \in \Gamma, \tag{29}$$

$$\sigma(\mathbf{x}) = f_2(\mathbf{x}) \text{ for } \mathbf{x} \in \Gamma \tag{30}$$

where Γ is the boundary enclosed the interested domain. The domain we consider is a square region and $x \in [0,1]$ and $y \in [0,1]$. The designed exact solutions for the field quantity and the conductivity are given as

$$u(x,y) = \frac{xy}{1+0.2x+0.4y+0.15xy} \tag{31}$$

and

$$\sigma(x,y) = (1+0.2x+0.4y+1.5xy)^2. \tag{32}$$

We use finite difference to discretize the domain by using a 31 by 31 mesh. Maximum 5% relative random errors are added in the data both in the surface field quantity as well as the surface conductivity. The parameters used for the double iteration process are: $\bar{\alpha}$=0.1, a_c =2.5, I_{max}=30000 and ε=10^{-5}. It reports that the process terminates for the number of steps for the outer loop is equal to 22 and the RMSE is lower than the requirement. The solutions for the field quantity and the conductivity are shown in Fig. 14 (a) and (b), respectively. The absolute error of the field quantity is illustrated in Fig. 15(a) while the relative error of the conductivity is illustrated in Fig. 15(b). The absolute error (AE) is defined as:

$$\text{AE:} = \|u_{num} - u_{true}\|$$

where u_{num} is the numerical solution and u_{true} is the analytic solution.

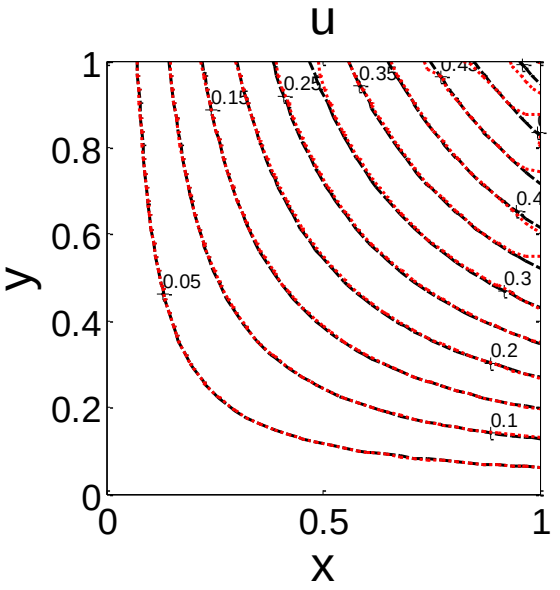
From these figures, we can say that the current approach gives accurate result and this method has good noise resistance.

[Example 6] In the following, a nonlinear inverse Cauchy problem for a quasi-linear PDE is considered as:

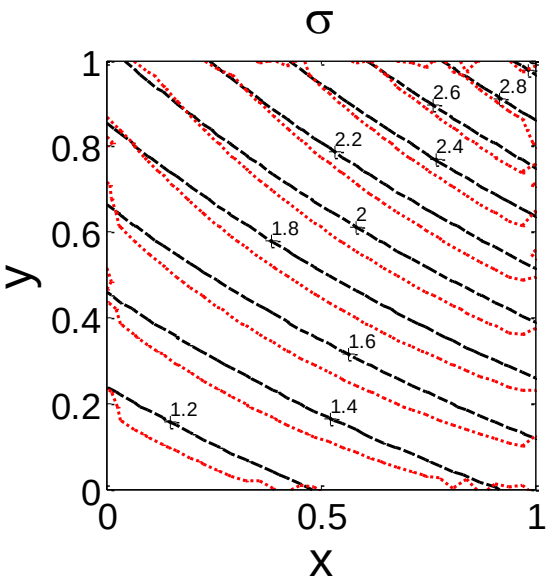
$$\nabla^2 u = 4u^3 \tag{33}$$

in a square region with $x \in [0,1]$ and $y \in [0,1]$.

The Cauchy boundary conditions are given on x=0 and x=1 while no boundary data are given on y=0 and y=1. Maximum 2% relative random errors are added into data. The designed solution is $u = \frac{1}{1+x+y}$. The forward problem of this nonlinear PDE can be found in [Liu (2008)]. The multiple quadrature radial basis functions are used to represent the field quantity. The radial basis function is written as $\varphi_{ij} = \sqrt{c^2 + r_{ij}^2}$ with $r_{ij} \equiv \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2}$ and c=1.5. Totally we arrange 30×30 mesh for the domain and boundary. It means we have 120 boundary points and 780 inner points. Parameters used for the double iteration process are: $\bar{\alpha}$=0.01, a_c =3.5, I_{max}=50000 and ε=10^{-4}. The initial guess for the weight of each radial basis function is equal to 0.001. After we obtain the weights for each radial basis functions, we use total 60x60 mesh to represent the solution. The contours for the exact solution and numerical solution are given in Fig. 16. Although this result is not so well as previous examples, for an ill-posed nonlinear system like this problem it is acceptable. The relative error percentage contour plot is illustrated in Fig. 17 and one can see that the maximum relative error percentage is about 5% while the maximum error in the boundary data is 2% which indicates that nonlinearity of the system may amplify the error. It should be mentioned here that in this case the process is terminated because the number of steps of inner loop exceeds I_{max}=50000.

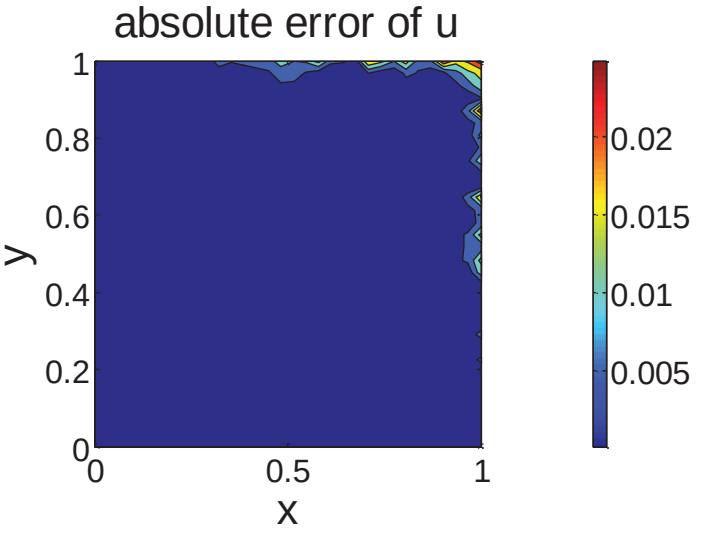


(a)

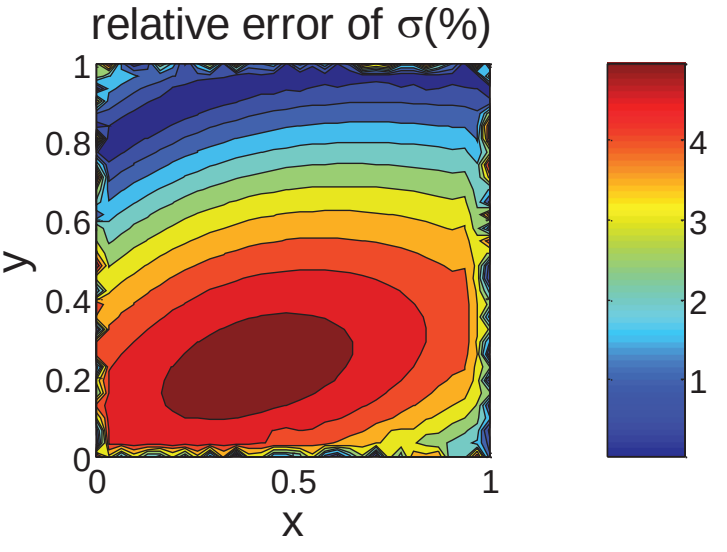


(b)

Figure 14: Numerical solutions for (a) the field quantity (b) the conductivity. (-+-: the exact solution; —: numerical solution)



(a)



(b)

Figure 15: Error distribution: (a) absolute error for the field quantity; (b)absolute relative error for the conductivity.

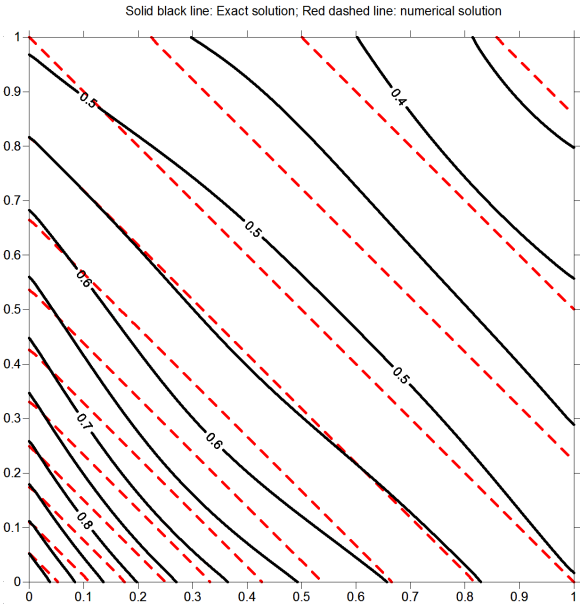


Figure 16: Comparison between the exact solution and the numerical solution.

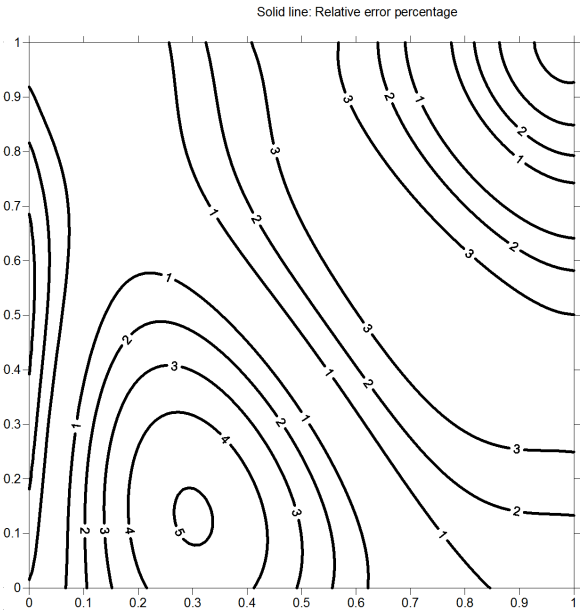


Figure 17: The absolute relative error percentage contour plot for example 6.

6 Conclusions

In this paper, a novel double iteration process for solving the nonlinear ill-posed system is proposed. The appropriate direction of evolution is determined from the inner loop which is based on the modified Tikhonov's regularization method. In order to avoid consuming too much computation effort, the whole process will be terminated when the number of iteration steps for the inner loop exceeds the maximum prescribed value. In such a case, it is said to further seek for the appropriate direction of evolution is computationally uneconomic and thus one should stop whole process. However, from numerical results we can observe that the numerical result is still acceptable. In other words, the proposed method is efficient and robust for solving the nonlinear ill-posed systems. Six examples are given to illustrate the validity for the proposed method.

Acknowledgement: The authors would like to express their thanks to H. F. Chan and C. L. Kuo for providing information and related source codes for our modification in example 4 and example 5.

References

Billups, S. C. (2002): A homotopy-based algorithm for mixed complementarity problems. *SIAM J. Optim.,* vol. 12 pp. 583-605.

Brown, K. M. (1973): *Computer oriented algorithms for solving systems of simultaneous nonlinear algebraic equations.* In Numerical Solution of Systems of Nonlinear Algebraic Equations, Byrne, G. D. and Hall C. A. Eds., pp. 281-348, Academic Press, New York.

Chan, H. F.; Fan, C. M. (2013): The modified collocation Trefftz method and exponentially convergent scalar homotopy algorithm for the inverse boundary determination problem for the biharmonic equation. *Journal of Mechanics,* vol. 29, pp. 363-372.

Chan, H. F.; Fan, C. M.; Yeih, W. (2011): Solution of inverse boundary optimization problem by Trefftz method and exponentially convergent scalar homotopy algorithm. *CMC: Computers, Materials, & Continua,* vol. 24, pp. 125-142.

Fu, Z. J.; Chen, W.; Zhang, C. Z. (2012): Boundary particle method for Cauchy inhomogeneous potential problems. *Inverse Problems in Science and Engineering,* vol. 20, no. 2, pp. 189-207.

Ku, C-Y; Yeih, W (2012): Dynamical Newton-like methods with adaptive stepsize for solving nonlinear algebraic equations. *CMC: Computers, Materials, & Continua,* vol. 31, pp. 173-200.

Ku, C-Y; Yeih, W; Liu, C-S (2011): Dynamical Newton-like methods for solving ill-conditioned systems of nonlinear equations with applications to boundary value problems. *CMES: Computer Modeling in Engineering & Sciences,* vol. 76, pp. 83-108.

Ku, C.-Y.; Yeih, W.; Liu, C.-S; Chi, C. C. (2009): Applications of the fictitious time integration method using a new time-like function. *CMES: Computer Modeling in Engineering & Sciences,* vol. 43, pp. 173-190.

Landweber, L. (1951): An iteration formula for Fredholm integral equations of the first kind. *Amer. J. Math.,* vol. 73, pp. 615–624.

Lin, J.; Chen, W.; Wang, F. (2011): A new investigation into regularization techniques for the method of fundamental solutions. *Mathematics and Computer in Simulation,* vol. 81, no. 6, pp. 1144-1152.

Liu, C.-S. (2008): A fictitious time integration method for two-dimensional quasi-linear elliptic boundary value problems. *CMES: Computer Modeling in Engineering & Sciences,* vol. 33, pp. 179-198.

Liu, C.-S. (2012): Optimally generalized regularization methods for solving linear inverse problems. *CMC: Computers, Materials, & Continua,* vol. 29, pp. 103-127.

Liu, C.-S. (2013): An optimal preconditioner with an alternate relaxation parameter used to solve ill-posed linear problems. *CMES: Computer Modeling in Engineering & Sciences,* vol. 92, pp. 241-269.

Liu, C.S.; Atluri, S. N. (2011a): An iterative algorithm for solving a system of nonlinear algebraic equations, F(x)=0, using the system of ODEs with an optimum in $\dot{\mathbf{x}} = \lambda \left[\alpha \mathbf{F} + (1-\alpha)\mathbf{B}^{\mathrm{T}}\mathbf{F}\right]; B_{ij} = \partial F_i / \partial x_j$. *CMES: Computer Modeling in Engineering & Sciences,* vol. 73, pp. 395431.

Liu, C.-S.; Atluri, S. N. (2011b): 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 & Sciences,* vol. 71, pp. 279-304.

Liu, C.-S.; Atluri, S. N. (2012): An iterative method using an optimal descent vector, for solving an ill-conditioned system BX = b, better and faster than the conjugate gradient method. *CMES: Computer Modeling in Engineering & Sciences,* vol. 80, pp. 275-298.

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

Polyanin, A. D.; Manzhirov, A. V. (2007): *Handbook of Integral Equations.* Second Edition, Caoman & Hall/CRC, Taylor & Francis Group, New York.

Ramm, A. G. (2007): Dynamical system methods for solving operator equations. *Mathematics in Science and Engineering,* vol. 208 Elsevier, Amsterdam, Netherlands.

Tikhonov, A. N.; Arsenin, V. Y. (1977): *Solution of Ill-posed Problems.* Washington: Winston & Sons.

Tjalling, J. Y. (1955): Historical development of the Newton-Raphson method. *SIAM Review,* vol. 37, pp. 531–551.

Wang, F.; Chen, W.; Ling, L.(2012): Combinations of the method of fundamental solutions for general inverse source identification problems. *Applied Mathematics and Computation,* vol. 19, no. 3, pp. 1173-1182.

Yeih, W.; Ku, C-Y; Liu, C.-S; Chan, I-Y (2013): A scalar homotopy method with optimal hybrid search directions for solving nonlinear algebraic equations. *CMES: Computer Modeling in Engineering & Sciences,* vol. 90 pp. 255-282.