Journal of Computational Mathematics
Vol.28, No.2, 2010, 235–260.

http://www.global-sci.org/jcm
doi:10.4208/jcm.2009.10-m2836

ON NEWTON-HSS METHODS FOR SYSTEMS OF NONLINEAR EQUATIONS WITH POSITIVE-DEFINITE JACOBIAN MATRICES*

Zhong-Zhi Bai
LSEC, ICMSEC, Academy of Mathematics and Systems Science, Chinese Academy of Sciences, Beijing 100190, China
Email: bzz@lsec.cc.ac.cn
Xue-Ping Guo
Department of Mathematics, East China Normal University, Shanghai 200062, China
Email: xpguo@math.ecnu.edu.cn

Abstract

The Hermitian and skew-Hermitian splitting (HSS) method is an unconditionally convergent iteration method for solving large sparse non-Hermitian positive definite system of linear equations. By making use of the HSS iteration as the inner solver for the Newton method, we establish a class of Newton-HSS methods for solving large sparse systems of nonlinear equations with positive definite Jacobian matrices at the solution points. For this class of inexact Newton methods, two types of local convergence theorems are proved under proper conditions, and numerical results are given to examine their feasibility and effectiveness. In addition, the advantages of the Newton-HSS methods over the Newton-USOR, the Newton-GMRES and the Newton-GCG methods are shown through solving systems of nonlinear equations arising from the finite difference discretization of a two-dimensional convection-diffusion equation perturbed by a nonlinear term. The numerical implementations also show that as preconditioners for the Newton-GMRES and the Newton-GCG methods the HSS iteration outperforms the USOR iteration in both computing time and iteration step.

Mathematics subject classification: 65F10, 65W05.
Key words: Systems of nonlinear equations, HSS iteration method, Newton method, Local convergence.

1. Introduction

Large sparse systems of nonlinear equations arise in many areas of scientific computing and engineering applications, e.g., in discretizations of nonlinear differential and integral equations, numerical optimization and so on; see [10, 26, 27] and references therein.

Let $F : \mathbb{D} \subset \mathbb{C}^n \to \mathbb{C}^n$ be a nonlinear and continuously differentiable mapping defined on the open convex domain $\mathbb{D}$ in the n-dimensional complex linear space $\mathbb{C}^n$, and consider systems of nonlinear equations of the form

$$F(x) = 0. \tag{1.1}$$

We assume that the Jacobian matrix of the nonlinear function $F(x)$ at the solution point $x_\star \in \mathbb{D}$, denoted as $F'(x_\star)$, is sparse, non-Hermitian, and positive definite. Here, the matrix $F'(x)$, for

* Received May 6, 2008 / Revised version received February 27, 2009 / Accepted June 8, 2009 / Published online December 21, 2009 /

$x \in \mathbb{D}$, is said to be positive definite if its Hermitian part

$$\mathcal{H}(F'(x)) := \frac{1}{2}(F'(x) + F'(x)^*)$$

is positive definite, where $F'(x)^*$ represents the conjugate transpose of $F'(x)$. For notational convenience, we also denote by

$$\mathcal{S}(F'(x)) := \frac{1}{2}(F'(x) - F'(x)^*)$$

the skew-Hermitian part of $F'(x)$; see [7, 12, 15, 18]. In this paper, we will study effective iteration methods and their convergence properties for solving this class of nonlinear systems.

The most classic and important solver for the system of nonlinear equations (1.1) may be the Newton method, which can be formulated as

$$x^{(k+1)} = x^{(k)} - F'(x^{(k)})^{-1}F(x^{(k)}), \qquad k = 0, 1, 2, \dots, \tag{1.2}$$

where $x^{(0)} \in \mathbb{D}$ is a given initial vector; see [11,26,27,29]. Obviously, at the k-th iteration step we need to solve the so-called Newton equation

$$F'(x^{(k)})s^{(k)} = -F(x^{(k)}), \quad \text{with} \quad x^{(k+1)} := x^{(k)} + s^{(k)}, \tag{1.3}$$

which is the dominant task in implementations of the Newton method. When the Jacobian matrix $F'(x)$ is large and sparse, iterative methods either of the splitting relaxation form (e.g., Gauss-Seidel, SOR [1)] and USOR [2)] ; see [19,26]) or of the Krylov subspace form (e.g., GMRES, BiCGSTAB and GCG [3)] ; see [4,25,28]) are often the methods of choice for effectively computing an approximation to the update vector $s^{(k)}$; see also [1, 2, 5, 6, 13]. This naturally results in the following inexact version of the Newton method for solving the system of nonlinear equations (1.1):

$$x^{(k+1)} = x^{(k)} + s^{(k)}, \quad \text{with} \quad F'(x^{(k)})s^{(k)} = -F(x^{(k)}) + r^{(k)}, \tag{1.4}$$

where $r^{(k)}$ is a residual yielded by the inner iteration due to the inexact solving; see [10,11,21,23]. Note that the convergence of the splitting relaxation methods is guaranteed only for Hermitian positive definite matrices or H-matrices, while this class of methods often requires much less computing operations at each iteration step and also much less computer storage than the Krylov subspace methods in actual implementations.

Recently, a Hermitian and skew-Hermitian splitting (HSS) iteration method was presented in [15] for solving large sparse system of linear equations with a non-Hermitian positive definite coefficient matrix, say $A \in \mathbb{C}^{n\times n}$; see also [12, 18]. Theoretical analysis has demonstrated that the HSS iteration method converges unconditionally to the exact solution, with the bound on the rate of convergence about the same as that of the conjugate gradient method when applied to the Hermitian matrix $\mathcal{H}(A) := \frac{1}{2}(A + A^*)$, and numerical experiments have shown that the HSS iteration method is very efficient and robust for solving non-Hermitian positive definite linear systems. Moreover, the HSS iteration method possesses a comparative memory requirement, but faster convergence rate, than the USOR iteration method, especially for matrices having strong skew-Hermitian parts.

[1)] SOR represents the successive overrelaxation method.
[2)] USOR represents the unsymmetric successive overrelaxation method.
[3)] GCG represents the generalized conjugate gradient method.

In this paper, instead of the classical splitting relaxation and the modern Krylov subspace iterations, we use the HSS iteration to solve approximately the Newton equation (1.3), obtaining a class of inexact Newton methods, called the Newton-HSS methods, for solving the system of nonlinear equations (1.1). Two types of local convergence theorems are established for the Newton-HSS methods, and numerical results are given to show their effectiveness and robustness. Moreover, numerical comparisons among the Newton-HSS, the Newton-USOR, the Newton-GMRES and the Newton-GCG methods show that the Newton-HSS method is much superior to the others in actual computations. It is also shown that as preconditioners for the Newton-GMRES and the Newton-GCG methods the HSS iteration outperforms the USOR iteration in both computing time and iteration step.

2. The Newton-HSS Methods

When $F : \mathbb{D} \subset \mathbb{C}^n \rightarrow \mathbb{C}^n$ is particularly a linear mapping, i.e., $F(x) = Ax - b$, with $A \in \mathbb{C}^{n\times n}$ a non-Hermitian positive definite matrix and $b \in \mathbb{C}^n$ a given right-hand-side vector, the system of nonlinear equations (1.1) reduces to the system of linear equations

$$Ax = b, \qquad A \in \mathbb{C}^{n\times n} \quad \text{and} \quad x, b \in \mathbb{C}^n. \tag{2.1}$$

Based on the Hermitian and skew-Hermitian (HS) splitting

$$A = H + S, \text{ with } H = \frac{1}{2}(A + A^*) \text{ and } S = \frac{1}{2}(A - A^*),$$

of the coefficient matrix A, Bai et al. [15] established the following HSS iteration method for solving the system of linear equations (2.1); see also [12, 18].

The HSS Iteration Method. Given an initial guess $x^{(0)} \in \mathbb{C}^n$, compute $x^{(\ell+1)}$ for $\ell = 0, 1, 2, \ldots$ using the following iteration scheme until $\{x^{(\ell)}\}$ satisfies the stopping criterion:

$$\begin{cases} (\alpha I + H)x^{(\ell+\frac{1}{2})} = (\alpha I - S)x^{(\ell)} + b, \\ (\alpha I + S)x^{(\ell+1)} = (\alpha I - H)x^{(\ell+\frac{1}{2})} + b, \end{cases}$$

where α is a given positive constant and I denotes the identity matrix.

In matrix-vector form, the above HSS iteration method can be equivalently rewritten as

$$\begin{aligned} x^{(\ell+1)} =& T(\alpha)x^{(\ell)} + G(\alpha)b \\ =& T(\alpha)^{\ell+1}x^{(0)} + \sum_{j=0}^{\ell} T(\alpha)^j G(\alpha)b, \qquad \ell = 0, 1, 2, \ldots, \end{aligned} \tag{2.2}$$

where

$$T(\alpha) = (\alpha I + S)^{-1}(\alpha I - H)(\alpha I + H)^{-1}(\alpha I - S) \tag{2.3}$$

and

$$G(\alpha) = 2\alpha(\alpha I + S)^{-1}(\alpha I + H)^{-1}. \tag{2.4}$$

Here, $T(\alpha)$ is the iteration matrix of the HSS method. In fact, (2.2) may also result from the splitting

$$A = B(\alpha) - C(\alpha)$$

of the coefficient matrix A, with

$$B(\alpha) = \frac{1}{2\alpha}(\alpha I + H)(\alpha I + S),$$
$$C(\alpha) = \frac{1}{2\alpha}(\alpha I - H)(\alpha I - S).$$

Notice that

$$(\alpha I - H)(\alpha I + H)^{-1} = (\alpha I + H)^{-1}(\alpha I - H).$$

It evidently holds that

$$T(\alpha) = B(\alpha)^{-1}C(\alpha) \quad \text{and} \quad G(\alpha) = B(\alpha)^{-1}.$$

The following theorem established in [15] describes the unconditional convergence property of the HSS iteration.

Theorem 2.1. *Let $A \in \mathbb{C}^{n\times n}$ be a positive definite matrix,*

$$H = \frac{1}{2}(A + A^*) \text{ and } S = \frac{1}{2}(A - A^*)$$

be its Hermitian and skew-Hermitian parts, respectively, and α be a positive constant. Then the spectral radius $\rho(T(\alpha))$ of the iteration matrix $T(\alpha)$ of the HSS iteration (*see* (2.3)) *is bounded by*

$$\sigma(\alpha) = \max_{\lambda_j \in \lambda(H)} \frac{|\alpha - \lambda_j|}{|\alpha + \lambda_j|},$$

where $\lambda(\cdot)$ represents the spectrum of the corresponding matrix. Consequently, we have

$$\rho(T(\alpha)) \le \sigma(\alpha) < 1, \qquad \forall \alpha > 0,$$

i.e., the HSS iteration converges to the exact solution $x_\star \in \mathbb{C}^n$ of the system of linear equations (2.1). *Moreover, if $\gamma_{\min}$ and $\gamma_{\max}$ are the lower and the upper bounds of the eigenvalues of the matrix H, respectively, then*

$$\tilde{\alpha} \equiv \arg\min_{\alpha} \left\{ \max_{\gamma_{\min} \le \lambda \le \gamma_{\max}} \left| \frac{\alpha - \lambda}{\alpha + \lambda} \right| \right\} = \sqrt{\gamma_{\min}\gamma_{\max}}$$

and

$$\sigma(\tilde{\alpha}) = \frac{\sqrt{\gamma_{\max}} - \sqrt{\gamma_{\min}}}{\sqrt{\gamma_{\max}} + \sqrt{\gamma_{\min}}} = \frac{\sqrt{\kappa(H)} - 1}{\sqrt{\kappa(H)} + 1},$$

where $\kappa(H)$ is the spectral condition number of H.

Based on the above preparation, we can now establish the Newton-HSS method for solving the system of nonlinear equations (1.1), which uses the Newton iteration (1.2) as the outer iteration and the HSS iteration as the inner iteration.

The Newton-HSS Method. Let $F : \mathbb{D} \subset \mathbb{C}^n \to \mathbb{C}^n$ be a continuously differentiable function with the positive-definite Jacobian matrix $F'(x)$ at any $x \in \mathbb{D}$, and

$$H(x) = \frac{1}{2}(F'(x) + F'(x)^*) \text{ and } S(x) = \frac{1}{2}(F'(x) - F'(x)^*)$$

be its Hermitian and skew-Hermitian parts, respectively. Given an initial guess $x^{(0)} \in \mathbb{D}$ and a sequence $\{l_k\}_{k=0}^{\infty}$ of positive integers, compute $x^{(k+1)}$ for $k = 0, 1, 2, \ldots$ using the following iteration scheme until $\{x^{(k)}\}$ satisfies the stopping criterion:

(a) Set $s^{(k,0)} := 0$;

(b) For $\ell = 0, 1, \dots, l_k - 1$, solve the following linear systems to obtain $s^{(k,\ell+1)}$:

$$\begin{cases} (\alpha I + H(x^{(k)}))s^{(k,\ell+\frac{1}{2})} = (\alpha I - S(x^{(k)}))s^{(k,\ell)} - F(x^{(k)}), \\ (\alpha I + S(x^{(k)}))s^{(k,\ell+1)} = (\alpha I - H(x^{(k)}))s^{(k,\ell+\frac{1}{2})} - F(x^{(k)}), \end{cases}$$

where α is a given positive constant;

(c) Set $x^{(k+1)} := x^{(k)} + s^{(k,l_k)}$.

In fact, the Newton-HSS method affords one feasible way of utilizing the HSS iteration to approximate solutions of the Newton equations in the Newton method for solving systems of nonlinear equations. In this case, we obtain a composite or multistep iteration scheme with the Newton method as the primary iteration and the HSS method as the secondary iteration.

By making use of (2.2), after straightforward operations we can obtain a uniform expression for $s^{(k,l_k)}$ as follows:

$$s^{(k,l_k)} = -\sum_{j=0}^{l_k-1} T(\alpha; x^{(k)})^j G(\alpha; x^{(k)}) F(x^{(k)}),$$

where

$$\begin{aligned} T(\alpha; x) &= (\alpha I + S(x))^{-1}(\alpha I - H(x))(\alpha I + H(x))^{-1}(\alpha I - S(x)) \\ G(\alpha; x) &= 2\alpha(\alpha I + S(x))^{-1}(\alpha I + H(x))^{-1}; \end{aligned} \tag{2.5}$$

see (2.3) and (2.4). It follows that the Newton-HSS method can be rewritten as the matrix-vector form

$$x^{(k+1)} = x^{(k)} - \sum_{j=0}^{l_k-1} T(\alpha; x^{(k)})^j G(\alpha; x^{(k)}) F(x^{(k)}), \qquad k = 0, 1, 2, \dots. \tag{2.6}$$

Define matrices

$$B(\alpha; x) = \frac{1}{2\alpha}(\alpha I + H(x))(\alpha I + S(x)), \tag{2.7a}$$

$$C(\alpha; x) = \frac{1}{2\alpha}(\alpha I - H(x))(\alpha I - S(x)). \tag{2.7b}$$

Then it holds that

$$F'(x) = B(\alpha; x) - C(\alpha; x) \tag{2.8}$$

is a splitting of the Jacobian matrix $F'(x)$,

$$\begin{aligned} &T(\alpha; x) = B(\alpha; x)^{-1} C(\alpha; x), \qquad B(\alpha; x) = G(\alpha; x)^{-1}, \\ &F'(x)^{-1} = (I - T(\alpha; x))^{-1} G(\alpha; x). \end{aligned} \tag{2.9}$$

Hence, from (2.6) we can equivalently express the Newton-HSS method as the alternative form

$$\begin{aligned} x^{(k+1)} &= x^{(k)} - \left(I - T(\alpha; x^{(k)})^{l_k}\right) F'(x^{(k)})^{-1} F(x^{(k)}) \\ &= x^{(k)} - F'(x^{(k)})^{-1}(F(x^{(k)}) - r(\alpha; x^{(k)}, l_k)), \qquad k = 0, 1, 2, \dots, \end{aligned} \tag{2.10}$$

which is evidently of the form of the inexact Newton method (1.4), with

$$r(\alpha; x, l) := F'(x)\, T(\alpha; x)^l F'(x)^{-1} F(x). \tag{2.11}$$

To end this section, we remark that the inner iterations of the Newton-Krylov methods, e.g., Newton-GMRES and Newton-GCG, may rely only on Jacobian-vector products, which can be approximated by a finite difference scheme and, hence, the actual Jacobian matrices need not be computed and stored in actual computations. The Newton-HSS method, however, requires the explicit Jacobian matrices. In addition, each inner HSS iteration requires solving two sub-systems of linear equations with respect to a shifted Hermitian and a shifted skew-Hermitian coefficient matrices and, hence, could be expensive if they are solved by direct methods. This seems a price that should be paid in using the Newton-HSS method to solve large sparse systems of nonlinear equations. A feasible remedy may be to solve these special sub-systems of linear equations inexactly by certain iteration methods, e.g., the conjugate gradient method or its variants, GMRES, BiCGSTAB, GCG and so on; see [17].

3. The Local Convergence Theory

First of all, for any $x \in \mathbb{C}^n$ and any $X \in \mathbb{C}^{n\times n}$ we define the vector norm $\|x\|$ by

$$\|x\| := \|(\alpha I + S(x_\star))x\|_2$$

and denote by

$$\|X\| := \|(\alpha I + S(x_\star))X(\alpha I + S(x_\star))^{-1}\|_2$$

the induced matrix norm, where $x_\star \in \mathbb{D}$ is a zero point of the nonlinear function $F : \mathbb{D} \subset \mathbb{C}^n \to \mathbb{C}^n$. Obviously, these norms are well defined as the matrix $\alpha I + S(x_\star)$ is positive definite and, hence, is nonsingular.

A mapping $F : \mathbb{D} \subset \mathbb{C}^n \to \mathbb{C}^n$ is *Gateaux*- (or G-) differentiable at an interior point x of $\mathbb{D}$ if there exists a linear operator $J \in \mathbb{C}^{n\times n}$ such that, for any $h \in \mathbb{C}^n$,

$$\lim_{t\to 0} \frac{1}{t}\|F(x+th) - F(x) - tJh\| = 0.$$

$F : \mathbb{D} \subset \mathbb{C}^n \to \mathbb{C}^n$ is said to be G-differentiable on an open set $\mathbb{D}_0 \subset \mathbb{D}$ if it is G-differentiable at any point in $\mathbb{D}_0$. Note that the above limit is independent of the particular norm on $\mathbb{C}^n$; that is, if F is G-differentiable in some norm, then it is G-differentiable in any norm.

The following perturbation lemma plays a fundamental role in the subsequent discussion; see Lemma 2.3.2 in [26, p. 45].

Lemma 3.1. [26] *Let $M, N \in \mathbb{C}^{n\times n}$ and assume that M is nonsingular, with $\|M^{-1}\| \le æ$. If $\|M - N\| \le \delta$ and $\delta æ < 1$, then N is also nonsingular, and*

$$\|N^{-1}\| \le \frac{æ}{1 - \delta æ}.$$

Theorem 11.1.5 in [26] gives a local convergence theory about an inexact Newton method that uses a general splitting iteration scheme as the inner solver. When this result is specified to the Newton-HSS method, we can immediately obtain the following local convergence theorem.

Theorem 3.1. *Let $F : \mathbb{D} \subset \mathbb{C}^n \to \mathbb{C}^n$ be G-differentiable on an open neighborhood $\mathbb{N}_0 \subset \mathbb{D}$ of a point $x_\star \in \mathbb{D}$ at which $F'(x)$ is continuous, positive definite, and $F(x_\star) = 0$. Suppose*

$$F'(x) = H(x) + S(x),$$

where

$$H(x) = \frac{1}{2}(F'(x) + F'(x)^*) \text{ and } S(x) = \frac{1}{2}(F'(x) - F'(x)^*)$$

are the Hermitian and the skew-Hermitian parts of the Jacobian matrix $F'(x)$, respectively. Then there exists an open neighborhood $\mathbb{N} \subset \mathbb{N}_0$ of $x_\star$ such that for any $x^{(0)} \in \mathbb{N}$ and any sequence of positive integers l_k, $k = 0, 1, 2, \ldots$, the iteration sequence $\{x^{(k)}\}_{k=0}^{\infty}$ generated by the Newton-HSS method is well-defined and convergent to $x_\star$. Moreover, it holds that

$$\limsup_{k\to\infty} \|x^{(k)} - x_\star\|^{\frac{1}{k}} \le \rho(T(\alpha; x_\star))^{l_o}, \quad \textit{with} \quad l_o = \liminf_{k\to\infty} l_k;$$

in particular, if $\lim_{k\to\infty} l_k = \infty$, then the rate of convergence is R-superlinear, i.e.,

$$\limsup_{k\to\infty} \|x^{(k)} - x_\star\|^{\frac{1}{k}} = 0.$$

Proof. It is straightforward from Theorem 11.1.5 in [26, page 350]. □

We remark that

$$\rho(T(\alpha; x_\star)) \le \max_{\lambda\in\lambda(H(x_\star))} \frac{|\alpha - \lambda|}{|\alpha + \lambda|} \equiv \sigma(\alpha; x_\star)$$

holds according to Theorem 2.1. In addition, when $l_k \equiv 1$, $k = 0, 1, 2, \ldots$, the Newton-HSS method reduces to the so-called one-step Newton-HSS method, which uses only one step of the HSS iteration to approximate the solution of the Newton equation at each step of the Newton method. Under the conditions of Theorem 3.1 the zero point $x_\star \in \mathbb{D}$ of the nonlinear function $F : \mathbb{D} \subset \mathbb{C}^n \to \mathbb{C}^n$ is an attraction point of the one-step Newton-HSS method, with the attraction factor being given by $\rho(T(\alpha; x_\star))$.

To know more exactly about the local convergence depending on the behaviour of the function F and the radius of the neighborhood $\mathbb{N}$ involved in Theorem 3.1, we establish the following local convergence theorem for the Newton-HSS method.

Theorem 3.2. *Let $F : \mathbb{D} \subset \mathbb{C}^n \to \mathbb{C}^n$ be G-differentiable on an open neighborhood $\mathbb{N}_0 \subset \mathbb{D}$ of a point $x_\star \in \mathbb{D}$ at which $F'(x)$ is continuous, positive definite, and $F(x_\star) = 0$. Suppose*

$$F'(x) = H(x) + S(x),$$

where

$$H(x) = \frac{1}{2}(F'(x) + F'(x)^*) \text{ and } S(x) = \frac{1}{2}(F'(x) - F'(x)^*)$$

are the Hermitian and the skew-Hermitian parts of the Jacobian matrix $F'(x)$, respectively. In addition, denote by $\mathbb{N}(x_\star, r)$ an open ball centered at $x_\star$ with radius r and assume the following conditions hold for all $x \in \mathbb{N}(x_\star, r) \subset \mathbb{N}_0$:

(A_1) (THE BOUNDED CONDITION) *there exist positive constants β and γ such that*

$$\max\{\|H(x_\star)\|, \quad \|S(x_\star)\|\} \le \beta \quad \textit{and} \quad \|F'(x_\star)^{-1}\| \le \gamma,$$

(A_2) (THE LIPSCHITZ CONDITION) *there exist nonnegative constants L_h and L_s such that*

$$\|H(x) - H(x_\star)\| \le L_h \|x - x_\star\|,$$
$$\|S(x) - S(x_\star)\| \le L_s \|x - x_\star\|.$$

Here $r \in (0, r_o)$, and r_o is defined by $r_o := \min_{1\le j\le 2}\{r_+^{(j)}\}$ and

$$r_+^{(1)} = \frac{\alpha+\beta}{L}\left(\sqrt{\frac{2\tau\alpha\theta}{\gamma(2+\tau\theta)(\alpha+\beta)^2}+1}-1\right),$$
$$r_+^{(2)} = \frac{1-2\beta\gamma[(\tau+1)\theta]^{l_o}}{3L\gamma},$$

with $L := L_h + L_s$, $l_o = \liminf_{k\to\infty} l_k$ satisfying

$$l_o > \lfloor -\frac{\ln(2\beta\gamma)}{\ln((\tau+1)\theta)} \rfloor, \tag{3.1}$$

where the symbol $\lfloor\cdot\rfloor$ is used to denote the smallest integer no less than the corresponding real number, $\tau \in (0, \frac{1-\theta}{\theta})$ a prescribed positive constant, and

$$\theta \equiv \theta(\alpha; x_\star) = \|T(\alpha; x_\star)\| \le \max_{\lambda\in\sigma(H(x_\star))} \frac{|\alpha-\lambda|}{|\alpha+\lambda|} \equiv \sigma(\alpha; x_\star).$$

Then, for any $x^{(0)} \in \mathbb{N}(x_\star, r)$ and any sequence $\{l_k\}_{k=0}^\infty$ of positive integers, the iteration sequence $\{x^{(k)}\}_{k=0}^\infty$ generated by the Newton-HSS method is well-defined and convergent to $x_\star$. Moreover, it holds that

$$\limsup_{k\to\infty} \|x^{(k)} - x_\star\|^{\frac{1}{k}} \le \theta^{l_o}.$$

Proof. Evidently, the bounded condition (A_1) directly implies the bounds

$$\begin{aligned}\|F'(x_\star)\| &= \|H(x_\star) + S(x_\star)\| \\ &\le \|H(x_\star)\| + \|S(x_\star)\| \le 2\beta\end{aligned} \tag{3.2}$$

and

$$\begin{aligned}\|B(\alpha; x_\star)^{-1}\| &= \|(I - T(\alpha; x_\star))F'(x_\star)^{-1}\| \\ &\le \|I - T(\alpha; x_\star)\|\|F'(x_\star)^{-1}\| \\ &\le (1 + \|T(\alpha; x_\star)\|)\|F'(x_\star)^{-1}\| \le 2\gamma.\end{aligned} \tag{3.3}$$

Here we have used the equalities (2.8)-(2.9) and the fact

$$\|T(\alpha; x_\star)\| \le \sigma(\alpha; x_\star) < 1;$$

see Theorem 2.1. In addition, the Lipschitz condition (A_2) also implies the Lipschitz continuity of the mapping $F' : \mathbb{D} \subset \mathbb{C}^n \to \mathbb{C}^{n\times n}$, i.e., it holds that

$$\begin{aligned}\|F'(x) - F'(x_\star)\| &\le \|H(x) - H(x_\star)\| + \|S(x) - S(x_\star)\| \\ &\le (L_h + L_s)\|x - x_\star\| = L\|x - x_\star\|.\end{aligned} \tag{3.4}$$

Therefore, it follows from the integral mean-value theorem that

$$\begin{aligned}
&\|F(x) - F(x_\star) - F'(x_\star)(x - x_\star)\| \\
=&\left\| \int_0^1 F'(x_\star + t(x - x_\star))(x - x_\star)dt - F'(x_\star)(x - x_\star) \right\| \\
\leq& \int_0^1 \|F'(x_\star + t(x - x_\star)) - F'(x_\star)\| \|x - x_\star\| dt \\
\leq& \int_0^1 Lt\|x - x_\star\|^2 dt = \frac{L}{2}\|x - x_\star\|^2
\end{aligned} \tag{3.5}$$

holds for all $x \in \mathbb{N}(x_\star, r)$. Because

$$\begin{aligned}
&H(x)S(x) - H(x_\star)S(x_\star) \\
=&[H(x) - H(x_\star)]S(x) + H(x_\star)[S(x) - S(x_\star)] \\
=&[H(x) - H(x_\star)][S(x) - S(x_\star)] + [H(x) - H(x_\star)]S(x_\star) + H(x_\star)[S(x) - S(x_\star)],
\end{aligned}$$

it follows from both (A_1) and (A_2) that for all $x \in \mathbb{N}(x_\star, r)$ we have

$$\begin{aligned}
&\|H(x)S(x) - H(x_\star)S(x_\star)\| \\
\leq&\|H(x) - H(x_\star)\|\|S(x) - S(x_\star)\| + \|H(x) - H(x_\star)\|\|S(x_\star)\| + \|H(x_\star)\|\|S(x) - S(x_\star)\| \\
\leq&L_h L_s\|x - x_\star\|^2 + \beta(L_h + L_s)\|x - x_\star\| \\
\leq&\frac{1}{2}(L_h + L_s)^2\|x - x_\star\|^2 + \beta(L_h + L_s)\|x - x_\star\| \\
=&\frac{1}{2}L^2\|x - x_\star\|^2 + L\beta\|x - x_\star\|.
\end{aligned} \tag{3.6}$$

Noticing that the equivalent expressions

$$B(\alpha; x) = \frac{1}{2\alpha}\Big(\alpha^2 I + \alpha F'(x) + H(x)S(x)\Big),$$

and

$$C(\alpha; x) = \frac{1}{2\alpha}\Big(\alpha^2 I - \alpha F'(x) + H(x)S(x)\Big)$$

about the matrices $B(\alpha; x)$ and $C(\alpha; x)$ defined in (2.7a-2.7b) straightforwardly lead to the equalities

$$B(\alpha; x) - B(\alpha; x_\star) = \frac{1}{2}\Big(F'(x) - F'(x_\star)\Big) + \frac{1}{2\alpha}\Big(H(x)S(x) - H(x_\star)S(x_\star)\Big),$$

and

$$C(\alpha; x) - C(\alpha; x_\star) = -\frac{1}{2}\Big(F'(x) - F'(x_\star)\Big) + \frac{1}{2\alpha}\Big(H(x)S(x) - H(x_\star)S(x_\star)\Big),$$

from (3.4) and (3.6) we can further obtain the estimates

$$\begin{aligned}
&\|B(\alpha; x) - B(\alpha; x_\star)\| \\
\leq&\frac{1}{2}\|F'(x) - F'(x_\star)\| + \frac{1}{2\alpha}\|H(x)S(x) - H(x_\star)S(x_\star)\| \\
\leq&\frac{L^2}{4\alpha}\|x - x_\star\|^2 + \frac{L(\alpha + \beta)}{2\alpha}\|x - x_\star\|
\end{aligned} \tag{3.7}$$

and

$$\|C(\alpha;x)-C(\alpha;x_\star)\| \le \frac{L^2}{4\alpha}\|x-x_\star\|^2+\frac{L(\alpha+\beta)}{2\alpha}\|x-x_\star\|. \tag{3.8}$$

Hence, by making use of the perturbation lemma, i.e., Lemma 3.1, it follows from (A_1) and (3.4) as well as (3.7) and (3.3) that

$$\|F'(x)^{-1}\| \le \frac{\gamma}{1-L\gamma\|x-x_\star\|} \tag{3.9}$$

and

$$\|B(\alpha;x)^{-1}\| \le \frac{4\alpha\gamma}{2\alpha-\gamma(L^2\|x-x_\star\|^2+2(\alpha+\beta)L\|x-x_\star\|)} \tag{3.10}$$

hold for all $x\in\mathbb{N}(x_\star,r)$, provided r is small enough such that $L\gamma\|x-x_\star\|<1$ and

$$\gamma\left(L^2\|x-x_\star\|^2+2(\alpha+\beta)L\|x-x_\star\|\right)<2\alpha.$$

Using (2.5), (2.8) and (2.9) immediately gives the equality

$$\begin{aligned}&T(\alpha;x)-T(\alpha;x_\star)\\ =&B(\alpha;x)^{-1}C(\alpha;x)-B(\alpha;x_\star)^{-1}C(\alpha;x_\star)\\ =&B(\alpha;x)^{-1}\left((C(\alpha;x)-C(\alpha;x_\star))-(B(\alpha;x)-B(\alpha;x_\star))T(\alpha;x_\star)\right).\end{aligned}$$

Based on (3.7), (3.8) and (3.10) we can obtain that

$$\begin{aligned}&\|T(\alpha;x)-T(\alpha;x_\star)\|\\ \le&\|B(\alpha;x)^{-1}\|[\|C(\alpha;x)-C(\alpha;x_\star)\|+\|B(\alpha;x)-B(\alpha;x_\star)\|\|T(\alpha;x_\star)\|]\\ \le&\frac{2\gamma\left(L^2\|x-x_\star\|^2+2(\alpha+\beta)L\|x-x_\star\|\right)}{2\alpha-\gamma\left(L^2\|x-x_\star\|^2+2(\alpha+\beta)L\|x-x_\star\|\right)}.\end{aligned}$$

Let us further restrict r so small that $L\gamma\|x-x_\star\|<1$ and

$$\gamma\left(L^2\|x-x_\star\|^2+2(\alpha+\beta)L\|x-x_\star\|\right)<\frac{2\tau\alpha\theta}{2+\tau\theta}.$$

Then it holds that

$$\frac{2\gamma\left(L^2\|x-x_\star\|^2+2(\alpha+\beta)L\|x-x_\star\|\right)}{2\alpha-\gamma\left(L^2\|x-x_\star\|^2+2(\alpha+\beta)L\|x-x_\star\|\right)}<\tau\theta$$

and, hence,

$$\begin{aligned}\|T(\alpha;x)\| &\le \|T(\alpha;x)-T(\alpha;x_\star)\|+\|T(\alpha;x_\star)\|\\ &\le \frac{2\gamma(L^2\|x-x_\star\|^2+2(\alpha+\beta)L\|x-x_\star\|)}{2\alpha-\gamma(L^2\|x-x_\star\|^2+2(\alpha+\beta)L\|x-x_\star\|)}+\theta\\ &\le(\tau+1)\theta.\end{aligned} \tag{3.11}$$

Now, we turn to estimate the error about the Newton-HSS iteration sequence $\{x^{(k)}\}_{k=0}^{\infty}$ defined by (2.10) and (2.11). Clearly, it holds that

$$
\begin{aligned}
& x^{(k+1)} - x_\star \\
=& x^{(k)} - x_\star - F'(x^{(k)})^{-1}F(x^{(k)}) + F'(x^{(k)})^{-1} r(\alpha; x^{(k)}, l_k) \\
=& - F'(x^{(k)})^{-1}\left(F(x^{(k)}) - F(x_\star) - F'(x^{(k)})(x^{(k)} - x_\star)\right) + \overline{r}(\alpha; x^{(k)}, l_k) \\
=& - F'(x^{(k)})^{-1}\left(F(x^{(k)}) - F(x_\star) - F'(x_\star)(x^{(k)} - x_\star)\right) \\
& + F'(x^{(k)})^{-1}\left(F'(x^{(k)}) - F'(x_\star\right)(x^{(k)} - x_\star) + \overline{r}(\alpha; x^{(k)}, l_k),
\end{aligned}
$$

where

$$
\begin{aligned}
& \overline{r}(\alpha; x, l) := F'(x)^{-1} r(\alpha; x, l) = T(\alpha; x)^l F'(x)^{-1} F(x) \\
=& T(\alpha; x)^l F'(x)^{-1}\left(F(x) - F(x_\star) - F'(x_\star)(x - x_\star)\right) \\
& + T(\alpha; x)^l F'(x)^{-1} F'(x_\star)(x - x_\star).
\end{aligned}
$$

Hence, by making use of (3.9), (3.2), (3.5) and (3.11) we can obtain

$$
\begin{aligned}
& \|\overline{r}(\alpha; x, l)\| \\
\leq & \|T(\alpha; x)^l\| \|F'(x)^{-1}\| \cdot \left(\|F(x) - F(x_\star) - F'(x_\star)(x - x_\star)\| + \|F'(x_\star)(x - x_\star)\|\right) \\
\leq & \frac{\gamma[(\tau+1)\theta]^l}{1 - L\gamma\|x - x_\star\|}\left(\frac{L}{2}\|x - x_\star\|^2 + 2\beta\|x - x_\star\|\right)
\end{aligned}
$$

and

$$
\begin{aligned}
& \|x^{(k+1)} - x_\star\| \\
\leq & \frac{3\gamma L}{2(1 - L\gamma\|x^{(k)} - x_\star\|)}\|x^{(k)} - x_\star\|^2 + \|\overline{r}(\alpha; x^{(k)}, l_k)\| \\
\leq & \frac{(3 + [(\tau+1)\theta]^{l_k})\gamma L}{2(1 - L\gamma\|x^{(k)} - x_\star\|)}\|x^{(k)} - x_\star\|^2 + \frac{2\beta\gamma[(\tau+1)\theta]^{l_k}}{1 - L\gamma\|x^{(k)} - x_\star\|}\|x^{(k)} - x_\star\| \\
\leq & \frac{2\gamma}{1 - L\gamma\|x^{(k)} - x_\star\|}\left(L\|x^{(k)} - x_\star\| + \beta[(\tau+1)\theta]^{l_k}\right)\|x^{(k)} - x_\star\| \\
:=& g(\|x^{(k)} - x_\star\|; l_k)\|x^{(k)} - x_\star\|,
\end{aligned}
$$

provided that

$$
\begin{aligned}
& L\gamma\|x^{(k)} - x_\star\| < 1, \\
& \gamma\left(L^2\|x^{(k)} - x_\star\|^2 + 2(\alpha+\beta)L\|x^{(k)} - x_\star\|\right) < \frac{2\tau\alpha\theta}{2 + \tau\theta}.
\end{aligned}
$$

Here, we have used the notation

$$
g(t; l) := \frac{2\gamma}{1 - L\gamma t}\left(Lt + \beta[(\tau+1)\theta]^l\right).
$$

By noticing that

$$
\begin{aligned}
g(\|x^{(k)} - x_\star\|; l_k) \leq & \frac{2\gamma}{1 - L\gamma\|x^{(k)} - x_\star\|}\left(L\|x^{(k)} - x_\star\| + \beta[(\tau+1)\theta]^{l_o}\right) \\
\leq & \frac{2\gamma}{1 - L\gamma r_o}(Lr_o + \beta[(\tau+1)\theta]^{l_o}) = g(r_o; l_o) < 1
\end{aligned}
$$

holds when (3.4) is satisfied and $x^{(k)} \in \mathbb{N}(x_\star, r_+^{(2)}) \subset \mathbb{N}(x_\star, r_o)$, we can further prove that $\{x^{(k)}\}_{k=0}^{\infty} \subset \mathbb{N}(x_\star, r)$ with the estimates

$$\|x^{(k+1)} - x_\star\| \le g(r_o; l_o)\|x^{(k)} - x_\star\|, \qquad k = 0, 1, 2, \ldots. \tag{3.12}$$

In fact, for $k = 0$ we have $\|x^{(0)} - x_\star\| < r < r_o$, as $x^{(0)} \in \mathbb{N}(x_\star, r)$. It follows from (3.12) that

$$\|x^{(1)} - x_\star\| \le g(r_o; l_o)\|x^{(0)} - x_\star\|,$$

which shows that (3.12) holds true for $k = 0$. In addition, we have

$$\|x^{(1)} - x_\star\| \le g(r_o; l_o)\|x^{(0)} - x_\star\| \le \|x^{(0)} - x_\star\| < r$$

and, hence, $x^{(1)} \in \mathbb{N}(x_\star, r)$. Suppose that $x^{(m)} \in \mathbb{N}(x_\star, r)$ and (3.12) is valid for some positive integer $k = m$. Then by making use of (3.12) again we can straightforwardly deduce the estimate

$$\|x^{(m+1)} - x_\star\| \le g(r_o; l_o)\|x^{(m)} - x_\star\|,$$

which shows that (3.12) holds true for $k = m + 1$, too. In addition, we have

$$\|x^{(m+1)} - x_\star\| \le g(r_o; l_o)\|x^{(m)} - x_\star\| \le \|x^{(m)} - x_\star\| < r$$

and, hence, $x^{(m+1)} \in \mathbb{N}(x_\star, r)$.

Now, the conclusion what we are proving follows as a direct corollary of (3.12) and Theorem 3.1. □

Theorem 3.1 shows that the attraction domain of the Newton-HSS method is $\mathbb{N}(x_\star, r_o)$. To obtain a large attraction domain, it is necessary that the positive constants L, β, γ and θ are small and the positive integer l_o is large. Roughly speaking, this implies that the function $F : \mathbb{D} \subset \mathbb{C}^n \to \mathbb{C}^n$ is mildly nonlinear, the Jacobian matrix $F'(x_\star)$ is well conditioned, and the inner iteration steps are reasonably large.

To our knowledge, there is no local convergence result of the type of Theorem 3.1 in the literature for an inexact Newton method using splitting iteration as its inner solver. Hence, Theorem 3.1 could be the first result on this topic.

4. Numerical Results

We consider the two-dimensional nonlinear convection-diffusion equation

$$\begin{cases} -(\mathrm{u_{xx}} + \mathrm{u_{yy}}) + q_1\mathrm{u_x} + q_2\mathrm{u_y} = -e^{\mathrm{u}}, & \text{for} \quad (\mathrm{x}, \mathrm{y}) \in \Omega, \\ \mathrm{u}(\mathrm{x}, \mathrm{y}) = 0, & \text{for} \quad (\mathrm{x}, \mathrm{y}) \in \partial\Omega, \end{cases} \tag{4.1}$$

where $\Omega = (0, 1) \times (0, 1)$, with $\partial\Omega$ its boundary, and q_1 and q_2 are positive constants used to measure the magnitudes of the convective terms; see [8, 9, 15]. By applying the centered finite difference scheme on the equidistant discretization grid with the stepsize $h = \frac{1}{N+1}$, we obtain the system of nonlinear equations (1.1) of the form

$$F(x) \equiv Mx + h^2\Phi(x) = 0,$$

where N is a prescribed positive integer,

$$M = T_{\mathrm{x}} \otimes I + I \otimes T_{\mathrm{y}},$$
$$\Phi(x) = (e^{x_1}, e^{x_2}, \ldots, e^{x_n})^T,$$

with $T_{\rm x}$ and $T_{\rm y}$ being tridiagonal matrices given by

$$T_{\rm x} = \text{tridiag}(-1-\text{Re}_1, 2, -1+\text{Re}_1) \quad \text{and} \quad T_{\rm y} = \text{tridiag}(-1-\text{Re}_2, 2, -1+\text{Re}_2).$$

Here, $\text{Re}_j = \frac{1}{2} q_j h$, $j = 1, 2$, $\text{Re} = \max\{\text{Re}_1, \text{Re}_2\}$ is the mesh Reynolds number, $\otimes$ the Kronecker product symbol, and $n = N \times N$; see [10, 22].

In actual computations, we take the positive constant q_2 to be $q_2 = \frac{1}{h}$ so that $\text{Re}_2 < 1$ is satisfied; see Section 5. In addition, the initial guess is chosen to be $x^{(0)} = 0$, the stopping criterion for the outer Newton iteration is set to be

$$\frac{\|F(x^{(k)})\|_2}{\|F(x^{(0)})\|_2} \le 10^{-6},$$

and that for the inner HSS iteration is set to be

$$\frac{\|F'(x^{(k)})s^{(k,l_k)} + F(x^{(k)})\|_2}{\|F(x^{(k)})\|_2} \le \eta,$$

where η is a prescribed tolerance for controlling the accuracy of the HSS iteration. The same stopping criterion is adopted for the inner iterations USOR, GMRES and GCG, too. The two sub-systems of linear equations with respect to the shifted Hermitian and the shifted skew-Hermitian coefficient matrices involved in the Newton-HSS iteration scheme are solved directly by making use of the sparse LU and Cholesky factorizations.

The Newton-HSS method is compared with the Newton-USOR, the Newton-GMRES and the Newton-GCG methods for different problem sizes $n = N \times N$, different quantities $q := q_1$ and different tolerances η, from aspects of the numbers of the outer, the inner and the total iteration steps (denoted as IT_{out}, IT_{int} and IT, respectively) and the total CPU time (denoted as CPU). Here IT_{int} denotes the average number of the inner iteration steps at each outer Newton iterate. Besides, the preconditioning effects of the HSS and the USOR iterations are examined when they are used to improve the numerical behaviours of the Newton-GMRES and the Newton-GCG methods.

In the implementations, we adopt the experimentally optimal parameters α for the Newton-HSS method and ω for the Newton-USOR method, which yield the least CPU times for these iteration methods, respectively; see Tables 4.1 and 4.2.

Table 4.1: The Optimal Values α for Newton-HSS Method

N	$q = 600$			$q = 800$			$q = 1000$		
	$\eta = 0.1$	$\eta = 0.2$	$\eta = 0.4$	$\eta = 0.1$	$\eta = 0.2$	$\eta = 0.4$	$\eta = 0.1$	$\eta = 0.2$	$\eta = 0.4$
30	3.0	2.7	2.9	1.1	1.2	1.1	1.1	1.1	1.4
40	1.3	1.2	1.3	1.2	1.1	1.3	1.4	1.2	1.3
50	1.6	1.5	1.8	1.2	1.5	1.2	1.2	1.2	1.3

Table 4.2: The Optimal Values ω for Newton-USOR Method

N	$q = 600$			$q = 800$			$q = 1000$		
	$\eta = 0.1$	$\eta = 0.2$	$\eta = 0.4$	$\eta = 0.1$	$\eta = 0.2$	$\eta = 0.4$	$\eta = 0.1$	$\eta = 0.2$	$\eta = 0.4$
30	0.3	0.3	0.3	0.2	0.2	0.2	0.2	0.2	0.2
40	0.3	0.3	0.3	0.3	0.3	0.3	0.2	0.2	0.2
50	0.4	0.4	0.4	0.3	0.3	0.3	0.2	0.2	0.2

Table 4.3: Numerical Results of Inexact Newton Methods for $\eta = 0.1$

N			30	40	50
$q = 600$	Newton-HSS	IT_{int}	6.0	5.7	5.5
		IT_{out}	6	6	6
		IT	36	34	33
		CPU	0.420	1.290	2.840
	Newton-USOR	IT_{int}	11.3	9.2	7.9
		IT_{out}	6	6	6
		IT	68	55	47
		CPU	0.790	2.130	4.740
	Newton-GMRES	IT_{int}	34.2	31.7	26.7
		IT_{out}	6	6	6
		IT	205	190	160
		CPU	0.430	1.380	2.960
	Newton-GCG	IT_{int}	44.5	47.2	54.5
		IT_{out}	6	6	6
		IT	267	283	327
		CPU	0.490	1.680	4.150
$q = 800$	Newton-HSS	IT_{int}	6.2	5.7	5.7
		IT_{out}	6	6	6
		IT	37	34	34
		CPU	0.470	1.280	2.870
	Newton-USOR	IT_{int}	13.0	12.3	9.7
		IT_{out}	6	6	6
		IT	78	74	58
		CPU	0.860	2.510	5.210
	Newton-GMRES	IT_{int}	40.0	34.2	33.3
		IT_{out}	6	6	6
		IT	240	205	200
		CPU	0.470	1.420	3.540
	Newton-GCG	IT_{int}	46.8	51.5	65.3
		IT_{out}	6	6	6
		IT	281	309	392
		CPU	0.520	1.790	4.810

For different inner tolerances η and problem parameters q, the results about IT_{out}, IT_{int}, IT and CPU are listed in the numerical tables corresponding to the referred inexact Newton methods and the preconditioned inexact Newton methods.

In Tables 4.3-4.6, we present the numerical results about the Newton method incorporated

Table 4.4: Numerical Results of Inexact Newton Methods for $\eta = 0.2$

N			30	40	50
$q = 600$	Newton-HSS	IT_{int}	4.4	4.6	4.6
		IT_{out}	8	7	7
		IT	35	32	32
		CPU	0.470	1.300	2.920
	Newton-USOR	IT_{int}	8.0	6.6	5.6
		IT_{out}	9	8	8
		IT	72	53	45
		CPU	0.950	2.370	5.230
	Newton-GMRES	IT_{int}	25.6	23.1	19.4
		IT_{out}	8	8	8
		IT	205	185	155
		CPU	0.440	1.400	3.100
	Newton-GCG	IT_{int}	27.2	27.0	30.2
		IT_{out}	9	9	9
		IT	245	243	272
		CPU	0.500	1.600	3.850
$q = 800$	Newton-HSS	IT_{int}	4.3	4.9	4.4
		IT_{out}	8	7	8
		IT	34	34	35
		CPU	0.500	1.360	3.180
	Newton-USOR	IT_{int}	9.4	9.0	6.8
		IT_{out}	9	8	9
		IT	85	72	61
		CPU	1.030	2.750	6.350
	Newton-GMRES	IT_{int}	30.0	25.6	24.4
		IT_{out}	8	8	8
		IT	240	205	195
		CPU	0.500	1.500	3.600
	Newton-GCG	IT_{int}	47.1	27.1	30.4
		IT_{out}	9	9	9
		IT	424	244	274
		CPU	0.750	1.630	3.890

with HSS, USOR, GMRES and GCG, corresponding to the inner tolerance $\eta = 0.1$, 0.2 and 0.4 and the problem parameter $q = 600$, 800 and 1000, respectively. From these tables, we can easily see that all these iteration methods can compute an approximate solution of the system of nonlinear equations. In particular, the Newton-HSS method considerably outperforms the

Table 4.5: Numerical Results of Inexact Newton Methods for $\eta = 0.4$

N			30	40	50
$q = 600$	Newton-HSS	IT_{int}	2.8	2.6	2.8
		IT_{out}	12	12	11
		IT	34	31	31
		CPU	0.550	1.610	3.480
	Newton-USOR	IT_{int}	4.7	4.0	3.7
		IT_{out}	14	13	12
		IT	66	52	44
		CPU	1.130	3.070	6.510
	Newton-GMRES	IT_{int}	15.4	14.6	12.5
		IT_{out}	13	12	12
		IT	200	175	150
		CPU	0.510	1.510	3.410
	Newton-GCG	IT_{int}	12.4	16.9	14.7
		IT_{out}	15	14	15
		IT	186	237	223
		CPU	0.520	1.840	4.000
$q = 800$	Newton-HSS	IT_{int}	2.8	2.8	2.8
		IT_{out}	12	12	12
		IT	33	33	33
		CPU	0.580	1.650	3.710
	Newton-USOR	IT_{int}	5.4	5.5	4.1
		IT_{out}	15	13	14
		IT	81	72	57
		CPU	1.280	3.460	7.820
	Newton-GMRES	IT_{int}	18.9	15.4	16.3
		IT_{out}	13	13	12
		IT	245	200	195
		CPU	0.570	1.690	3.990
	Newton-GCG	IT_{int}	21.3	14.9	16.7
		IT_{out}	15	15	15
		IT	319	223	250
		CPU	0.700	1.810	4.270

Newton-USOR, the Newton-GMRES and the Newton-GCG methods for all the tested cases, as it has the least iteration step and CPU time, which are much less than those of the others.

In Tables 4.7-4.10, we present the numerical results about the Newton-GMRES and the Newton-GCG methods preconditioned by HSS and USOR, corresponding to the inner tolerance

Table 4.6: Numerical Results of Inexact Newton Methods for $q = 1000$

N			30	40	50
$\eta = 0.2$	Newton-HSS	IT_{int}	4.5	4.4	4.4
		IT_{out}	8	8	8
		IT	36	35	35
		CPU	0.500	1.450	3.210
	Newton-USOR	IT_{int}	12.4	9.6	9.3
		IT_{out}	9	9	9
		IT	112	86	84
		CPU	1.220	3.180	7.510
	Newton-GMRES	IT_{int}	36.3	29.4	28.1
		IT_{out}	8	8	8
		IT	290	235	225
		CPU	0.570	1.650	4.000
	Newton-GCG	IT_{int}	39.7	33.9	32.9
		IT_{out}	9	9	9
		IT	357	305	296
		CPU	0.660	1.860	4.110
$\eta = 0.4$	Newton-HSS	IT_{int}	3.5	3.1	2.9
		IT_{out}	11	11	12
		IT	38	34	35
		CPU	0.580	1.610	3.780
	Newton-USOR	IT_{int}	7.3	5.6	5.6
		IT_{out}	15	14	15
		IT	109	78	79
		CPU	1.480	3.720	8.920
	Newton-GMRES	IT_{int}	21.4	19.2	17.9
		IT_{out}	14	12	12
		IT	300	230	215
		CPU	0.670	1.780	4.250
	Newton-GCG	IT_{int}	21.1	15.7	22.0
		IT_{out}	15	15	15
		IT	317	235	330
		CPU	0.700	1.850	5.030

$\eta = 0.1$, 0.2 and 0.4 and the problem parameter $q = 600$, 800 and 1000, respectively. From these tables, we can easily see that all these iteration methods can compute an approximate solution of the system of nonlinear equations. In particular, as preconditioners the HSS iteration is much more effective than the USOR iteration for all the tested cases in the sense of iteration step and computing time, when they are used to improve the numerical behaviours of the

Table 4.7: Numerical Results of Preconditioned Inexact Newton Methods for $\eta = 0.1$

N				30	40	50
$q = 600$	GMRES	HSS	IT_{int}	5.0	8.8	7.5
			IT_{out}	4	4	4
			IT	20	35	30
			CPU	0.320	1.040	2.230
		USOR	IT_{int}	5.0	8.0	8.8
			IT_{out}	4	5	4
			IT	20	40	35
			CPU	0.380	1.730	3.570
	GCG	HSS	IT_{int}	3.2	5.3	6.2
			IT_{out}	6	6	6
			IT	19	32	37
			CPU	0.340	1.090	2.580
		USOR	IT_{int}	3.8	13.3	9.3
			IT_{out}	5	6	6
			IT	19	80	56
			CPU	0.400	2.240	4.460
$q = 800$	GMRES	HSS	IT_{int}	8.0	8.8	8.8
			IT_{out}	5	4	4
			IT	40	35	35
			CPU	0.450	1.070	2.400
		USOR	IT_{int}	9.0	5.0	7.0
			IT_{out}	5	4	5
			IT	45	20	35
			CPU	0.600	1.140	3.890
	GCG	HSS	IT_{int}	6.3	5.8	6.2
			IT_{out}	6	6	6
			IT	38	35	37
			CPU	0.420	1.130	2.640
		USOR	IT_{int}	8.2	3.8	9.2
			IT_{out}	6	6	6
			IT	49	23	55
			CPU	0.570	1.350	4.420

Newton-GMRES and the Newton-GCG methods.

Table 4.8: Numerical Results of Preconditioned Inexact Newton Methods for $\eta = 0.2$

N				30	40	50
$q = 600$	GMRES	HSS	IT_{int}	5.0	5.0	5.0
			IT_{out}	4	7	6
			IT	20	35	30
			CPU	0.300	1.230	2.560
		USOR	IT_{int}	5.0	6.7	7.0
			IT_{out}	4	6	5
			IT	20	40	35
			CPU	0.370	1.870	3.900
	GCG	HSS	IT_{int}	2.5	4.4	4.1
			IT_{out}	8	8	8
			IT	20	35	33
			CPU	0.370	1.260	2.840
		USOR	IT_{int}	2.5	8.3	4.0
			IT_{out}	8	8	8
			IT	20	66	32
			CPU	0.500	2.290	4.260
$q = 800$	GMRES	HSS	IT_{int}	5.0	5.0	5.0
			IT_{out}	7	7	6
			IT	35	35	30
			CPU	0.450	1.260	2.560
		USOR	IT_{int}	8.3	5.0	7.0
			IT_{out}	6	4	5
			IT	50	20	35
			CPU	0.680	1.140	3.910
	GCG	HSS	IT_{int}	3.9	4.8	3.9
			IT_{out}	8	8	8
			IT	31	38	31
			CPU	0.440	1.300	2.810
		USOR	IT_{int}	6.0	3.0	5.5
			IT_{out}	8	7	8
			IT	48	21	44
			CPU	0.650	1.450	4.710

5. Remarks

In this section, we make remarks about stable discretizations of convection-diffusion equations and globally convergent variants of the Newton-HSS method.

Table 4.9: Numerical Results of Preconditioned Inexact Newton Methods for $\eta = 0.4$

N				30	40	50
$q = 600$	GMRES	HSS	IT_{int}	5.0	5.0	5.0
			IT_{out}	4	6	5
			IT	20	30	25
			CPU	0.300	1.110	2.290
		USOR	IT_{int}	5.0	5.7	5.0
			IT_{out}	4	7	7
			IT	20	40	35
			CPU	0.370	2.010	4.540
	GCG	HSS	IT_{int}	1.7	2.8	2.5
			IT_{out}	10	11	11
			IT	17	31	27
			CPU	0.390	1.400	3.190
		USOR	IT_{int}	1.8	3.6	2.6
			IT_{out}	11	13	11
			IT	20	47	29
			CPU	0.650	2.710	5.180
$q = 800$	GMRES	HSS	IT_{int}	5.0	5.0	5.0
			IT_{out}	7	6	7
			IT	35	30	35
			CPU	0.460	1.130	2.850
		USOR	IT_{int}	5.0	5.0	5.8
			IT_{out}	9	4	6
			IT	45	20	35
			CPU	0.770	1.140	4.230
	GCG	HSS	IT_{int}	2.7	2.5	2.8
			IT_{out}	13	13	13
			IT	35	33	36
			CPU	0.560	1.560	3.700
		USOR	IT_{int}	3.6	2.0	3.8
			IT_{out}	13	10	11
			IT	47	20	42
			CPU	0.880	1.860	5.650

5.1. The Stable Discretizations

When solving partial differential equations it is important to use a stable discretization, as otherwise the discrete solution may not converge and, normally, the approximate solution will be contaminated with noise, i.e., shows an oscillating behaviour.

Table 4.10: Numerical Results of Preconditioned Inexact Newton Methods for $q = 1000$

N				30	40	50
$\eta = 0.2$	GMRES	HSS	IT_{int}	5.0	5.0	5.0
			IT_{out}	7	7	7
			IT	35	35	35
			CPU	0.450	1.280	2.890
		USOR	IT_{int}	5.0	8.3	9.2
			IT_{out}	5	6	6
			IT	25	50	55
			CPU	0.440	2.090	5.320
	GCG	HSS	IT_{int}	4.3	4.1	4.5
			IT_{out}	8	8	8
			IT	34	33	36
			CPU	0.450	1.260	2.960
		USOR	IT_{int}	2.9	8.1	17.9
			IT_{out}	8	8	9
			IT	23	65	161
			CPU	0.520	2.300	9.380
$\eta = 0.4$	GMRES	HSS	IT_{int}	5.0	5.0	5.0
			IT_{out}	6	6	6
			IT	30	30	30
			CPU	0.410	1.150	2.600
		USOR	IT_{int}	5.0	6.3	6.9
			IT_{out}	5	8	8
			IT	25	50	55
			CPU	0.450	2.360	5.970
	GCG	HSS	IT_{int}	2.3	2.6	2.8
			IT_{out}	12	12	12
			IT	27	31	33
			CPU	0.490	1.480	3.490
		USOR	IT_{int}	2.0	5.0	8.2
			IT_{out}	11	14	13
			IT	22	70	107
			CPU	0.650	3.210	8.750

For simplicity, we shall here consider only the linear convection-diffusion problems due to the nice property of the nonlinear term e^{u} involved in the nonlinear convection-diffusion equation (4.1). As is well known, when one uses a central difference approximation for convection-diffusion problems, the solution is normally heavily contaminated with noise, when the diffusion parameter $\nu < h$, where h is an average stepsize of the mesh used, and when the solution has a boundary or interior layer. However, it turns out that the noise contaminates essentially only the even points (starting the ordering from the first point next to the boundary layer) but not the odd numbered points. This can be explained if one uses an odd-even reordering of the equations and unknowns to form a two-by-two block linear system

$$A_h = \begin{bmatrix} D_1 & E_{12} \\ E_{21} & D_2 \end{bmatrix},$$

where the first block row corresponds to the even-numbered equations. Eliminating these, one gets a system matrix

$$D_2 - E_{21} D_1^{-1} E_{12}$$

in the odd-numbered unknowns, which, under certain conditions, turns out to be an M-matrix and is hence stable.

For example, after discretization of the one-dimensional boundary value problem

$$\begin{cases} -\nu \mathrm{u}'' + q\mathrm{u}' = 0, & 0 < \mathrm{x} < 1, \\ \mathrm{u}(0) = \mathrm{u}(1) = 1, & \end{cases} \qquad \nu > 0,$$

on a uniform mesh Ω_h with spacing h and constants ν and q, the central difference matrix takes the form

$$A_h = -\nu h^{-2} \cdot \mathrm{tridiag}(-1 - P_e, 2, -1 + P_e),$$

where $P_e = \frac{qh}{2\nu}$ is the Peclet number. If $P_e \leq 1$, then A_h is an M-matrix, i.e., in particular a monotone matrix ($A_h v \geq 0$ for any real vector $v \geq 0$). But this does not hold if $P_e > 1$. Using the odd-even reordering and elimination of the even-ordered equations results in a new difference approximation where

$$D_i = \mathrm{diag}(2), \quad E_{12} = \mathrm{tridiag}(-1 - P_e, -1 + P_e, 0), \quad E_{21} = \mathrm{tridiag}(0, -1 - P_e, -1 + P_e)$$

and the reduced linear system takes the form

$$D_2 - E_{12} D_1^{-1} E_{21} = -\frac{1}{2} \nu h^{-2} \cdot \mathrm{tridiag}(-(1 + P_e)^2, 2(1 + P_e^2), -(1 - P_e)^2)$$

for the odd-ordered points. Note that this is an M-matrix for all values of P_e. The reduced linear system is, in fact, equivalent to a central difference approximation to

$$-\nu(1 + P_e^2)\mathrm{u}'' + q\mathrm{u}' = 0 \tag{5.1}$$

on the double-spaced mesh, say, Ω_{2h}. Hence, the approximation on the originally odd-numbered points do not show any unphysical wiggles.

The solution to (5.1) can be quite acceptable also for small values ν if there are no layers, but if layers are present it shows too much dispersion (smearing) of the layers. However, if one resolves the layers by using a sufficiently fine mesh in the layers, then the global solution becomes quite acceptable. Actually, we could better have added an (even smaller) amount of artificial diffusion, $\nu(\frac{P_e}{2})^2$, directly to the original equation and used the central difference approximation method on this, if we are content with this type of monotone (and second order correct !) but heavily smeared approximations. Actually, there is a simple "trick" to improve the solution of the central difference approximation substantially. We then interpolate the solution given at the even points to the odd points and then take the arithmetic average of the original solution at an odd point and the interpolated value at this point.

The reduction method to get a monotone operator can be generalized to the operator

$$-\nu\Delta \mathrm{u} + q_1 \mathrm{u_x} + q_2 \mathrm{u_y}$$

in a rectangular domain if $q_1 > 0$ and $P_{e_2} = \frac{|q_2|h}{2\nu} < 1$. That the reduced equations give a monotone operator even for problems with variable coefficients has been shown earlier in [8]; see also [9, 22].

Consider now the upwind difference approximation where we use a backward difference, i.e.,

$$q_1 \frac{\partial \mathrm{u}}{\partial \mathrm{x}} \approx q_1(\mathrm{x},\mathrm{y}) \cdot \frac{\mathrm{u}(\mathrm{x},\mathrm{y}) - \mathrm{u}(\mathrm{x}-h,\mathrm{y})}{h}, \quad \text{if} \quad q_1 > 0,$$

or a forward difference approximation if $q_1 < 0$, and corresponding approximations for $q_2 \frac{\partial \mathrm{u}}{\partial \mathrm{y}}$. Note then that for sufficiently regular solutions,

$$q_1(\mathrm{x},\mathrm{y}) \cdot \frac{\mathrm{u}(\mathrm{x},\mathrm{y}) - \mathrm{u}(\mathrm{x}-h,\mathrm{y})}{h} = q_1 \mathrm{u_x}(\mathrm{x},\mathrm{y}) - \frac{h}{2} q_1 \mathrm{u_{xx}}(\mathrm{x},\mathrm{y}) + \mathcal{O}(h^2).$$

Hence, the upwind scheme is similar to the use of central differences on the equation

$$-\nu(1 + P_{e_1})\mathrm{u_{xx}} - \nu(1 + P_{e_2})\mathrm{u_{yy}} + q_1 \mathrm{u_x} + q_2 \mathrm{u_y} = g(\mathrm{x},\mathrm{y}),$$

where $P_{e_i} = \frac{|q_i|h}{2\nu}$, $i = 1, 2$.

Since we have here added artificial diffusion of order νP_{e_i}, $i = 1, 2$, this scheme is only first-order accurate. Furthermore, it has dispersion behaviour. The advantage with it is that it gives an M-matrix for all values of P_{e_i}, i.e., in particular a monotone approximation, for which there can appear no unphysical wiggles.

Using the classical barrier lemma, valid for monotone operators, we can prove a discretization error estimate in supreme norm of first order accuracy for the upwind difference method. If $P_{e_i} \le 1$, this can also be proved for the central difference method (of second order accuracy).

Clearly, the symmetric part is relatively strong for M-matrices. The above observations have been done earlier in [8]; see also [7].

5.2. The Damped Newton-HSS Method

Theorems 3.1 and 3.2 have shown that the Newton-HSS method has local convergence property. In actual applications, however, an iteration scheme of global convergence is often much more important and practical. Fortunately, we can modify the Newton-HSS method to obtain a globally convergent nonlinear iteration method by simply introducing a damping factor, say, t. This iteration method is termed as the damped Newton-HSS method and is algorithmically described as follows.

The Damped Newton-HSS Method. Let $F : \mathbb{D} \subset \mathbb{C}^n \to \mathbb{C}^n$ be a continuously differentiable function with the positive-definite Jacobian matrix $F'(x)$ at any $x \in \mathbb{D}$, and $H(x) = \frac{1}{2}(F'(x) + F'(x)^*)$ and $S(x) = \frac{1}{2}(F'(x) - F'(x)^*)$ be its Hermitian and skew-Hermitian parts, respectively. Given an initial guess $x^{(0)} \in \mathbb{D}$, a sequence $\{t_k\}_{k=0}^{\infty}$ of positive reals and a sequence $\{l_k\}_{k=0}^{\infty}$ of positive integers, compute $x^{(k+1)}$ for $k = 0, 1, 2, \ldots$ using the following iteration scheme until $\{x^{(k)}\}$ satisfies the stopping criterion:

(a) Set $s^{(k,0)} := 0$;

(b) For $\ell = 0, 1, \ldots, l_k - 1$, solve the following linear systems to obtain $s^{(k,\ell+1)}$:

$$\begin{cases} (\alpha I + H(x^{(k)}))s^{(k,\ell+\frac{1}{2})} = (\alpha I - S(x^{(k)}))s^{(k,\ell)} - F(x^{(k)}), \\ (\alpha I + S(x^{(k)}))s^{(k,\ell+1)} = (\alpha I - H(x^{(k)}))s^{(k,\ell+\frac{1}{2})} - F(x^{(k)}), \end{cases}$$

where α is a given positive constant;

(c) Set $x^{(k+1)} := x^{(k)} + t_k s^{(k,l_k)}$.

By suitably choosing the sequence $\{t_k\}_{k=0}^{\infty}$ of the stepsizes and the stopping criterion of the inner HSS iterations, we can make the damped Newton-HSS iteration sequence $\{x^{(k)}\}_{k=0}^{\infty}$ satisfy the norm-reducing requirement $\|F(x^{(k+1)})\| \leq \eta_k \|F(x^{(k)})\|$, $k = 0, 1, 2, \ldots$, in some norm and, therefore, obtain the global convergence of the damped Newton-HSS method; see [1,2,13]. Here, $\{\eta_k\}_{k=0}^{\infty}$ is a forcing sequence used to control the inner HSS iterations. For more details about strategies and techniques on treatments about the global convergence of the approximate or the inexact Newton methods, we refer to [3,20,24].

6. Conclusions

For large sparse systems of nonlinear equations where the Hermitian parts of the corresponding Jacobian matrices are positive definite, we have established a class of inner-outer iteration schemes, called the Newton-HSS iteration methods, and proved its local convergence property. It has been demonstrated by numerical examples that the Newton-HSS iteration method can outperform the Newton-USOR, the Newton-GMRES and the Newton-GCG iteration methods; and as preconditioners, the HSS iteration is superior to the USOR iteration.

Amazingly, this holds not only with respect to computing time but also with respect to number of iterations. A typical application of the method is for nonlinear convection-diffusion equations.

For singularly perturbed problems the method as given may become inapplicable as it would require excessively small values of h. However, as shown in [9], one can then use a defect-correction or iterative refinement method (normally involving just two or three steps) where the correction is based on a stable, upwind type difference operator for which the Hermitian part of the Jacobian matrix is sufficiently dominating. In this way, the prescribed method becomes applicable also for singularly perturbed differential operators.

At last, we should mention that the Newton-HSS methods are only a special case of the general principle of combining nonlinear iterative methods with linear iterations in order to form composite or multistep iteration methods. The Newton method itself may be replaced as the primary iteration by, for example, any of the discretized Newton, the secant or the Steffensen methods. Hence, we can correspondingly obtain the discretized Newton-HSS, the secant-HSS or the Steffensen-HSS methods, respectively. Alternatively, the HSS iteration itself may be replaced as the secondary iteration by, for example, any of the NSS[1)] [12,16], the PSS[2)] [12,14]

---

1) NSS represents the normal and skew-Hermitian splitting.

2) PSS represents the positive-definite and skew-Hermitian splitting.

or the BTSS[3)] [12,14] iterations and, thereby, we can correspondingly obtain the Newton-NSS, the Newton-PSS and the Newton-BTSS iteration methods, respectively. Theoretical analyses and numerical implementations of these composite iteration methods are interesting topics in future study.

Acknowledgements. The authors thank Mr. Xi Yang for helping on running the numerical results. They are also very much indebted to the referees for providing useful comments and suggestions, which greatly improved the original manuscript of this paper. The research is supported by the National Basic Research Program (No. 2005CB321702), the China Outstanding Young Scientist Foundation (No. 10525102) and the National Natural Science Foundation (No. 10471146), (No. 10571059 and No. 10571060), P.R. China.

References

[1] H.B. An and Z.Z. Bai, NGLM: A globally convergent Newton-GMRES method, *Math. Numer. Sin.*, **27** (2005), 151-174. (In Chinese).

[2] H.B. An and Z.Z. Bai, A globally convergent Newton-GMRES method for large sparse systems of nonlinear equations, *Appl. Numer. Math.*, **57** (2007), 235-252.

[3] O. Axelsson, On global convergence of iterative methods, in Iterative Solution of Nonlinear Systems of Equations, Lecture Notes in Mathematics, 953, pp. 1-19, *Springer-Verlag*, Berlin and New York, 1982.

[4] O. Axelsson, A generalized conjugate gradient, least square method, *Numer. Math.*, **51** (1987), 209-227.

[5] O. Axelsson, A restarted version of a generalized preconditioned conjugate gradient method, *Commun. Appl. Numer. Methods*, **4** (1988), 521-530.

[6] O. Axelsson, Iterative Solution Methods, *Cambridge University Press*, Cambridge, 1994.

[7] O. Axelsson, Z.Z. Bai and S.X. Qiu, A class of nested iteration schemes for linear systems with a coefficient matrix with a dominant positive definite symmetric part, *Numer. Algorithms*, **35** (2004), 351-372.

[8] O. Axelsson and G.F. Carey, On the numerical solution of two-point singularly perturbed boundary value problems, *Comput. Method. Appl. Mech. Engrg.*, **50** (1985), 217-229.

[9] O. Axelsson and M. Nikolova, Avoiding slave points in adaptive refinement procedures for convection-diffusion problems in 2D, *Computing*, **61** (1998), 331-357.

[10] Z.Z. Bai, A class of two-stage iterative methods for systems of weakly nonlinear equations, *Numer. Algorithms*, **14** (1997), 295–319.

[11] Z.Z. Bai, On the convergence of parallel chaotic nonlinear multisplitting Newton-type methods, *J. Comput. Appl. Math.*, **80** (1997), 317-334.

[12] Z.Z. Bai, Splitting iteration methods for non-Hermitian positive definite systems of linear equations, *Hokkaido Math. J.*, **36** (2007), 801-814.

[13] Z.Z. Bai and H.B. An, On efficient variants and global convergence of the Newton-GMRES method, *J. Numer. Methods Computer Appl.*, **26** (2005), 291-300. (In Chinese)

[14] Z.Z. Bai, G.H. Golub, L.Z. Lu and J.F. Yin, Block triangular and skew-Hermitian splitting methods for positive-definite linear systems, *SIAM J. Sci. Comput.*, **26** (2005), 844-863.

[15] Z.Z. Bai, G.H. Golub and M.K. Ng, Hermitian and skew-Hermitian splitting methods for non-Hermitian positive definite linear systems, *SIAM J. Matrix Anal. Appl.*, **24** (2003), 603-626.

[16] Z.Z. Bai, G.H. Golub and M.K. Ng, On successive overrelaxation acceleration of the Hermitian and skew-Hermitian splitting iterations, *Numer. Linear Algebra Appl.*, **14** (2007), 319-335.

[3)] BTSS represents the block triangular and skew-Hermitian splitting.

[17] Z.Z. Bai, G.H. Golub and M.K. Ng, On inexact Hermitian and skew-Hermitian splitting methods for non-Hermitian positive definite linear systems, *Linear Algebra Appl.*, **428** (2008), 413-440.

[18] Z.Z. Bai, G.H. Golub and J.Y. Pan, Preconditioned Hermitian and skew-Hermitian splitting methods for non-Hermitian positive semidefinite linear systems, *Numer. Math.*, **98** (2004), 1-32.

[19] Z.Z. Bai, J.C. Sun and D.R. Wang, A unified framework for the construction of various matrix multisplitting iterative methods for large sparse system of linear equations, *Comput. Math. Appl.*, **32** (1996), 51-76.

[20] R.E. Bank and D.J. Rose, Global approximate Newton methods, *Numer. Math.*, **37** (1981), 279-295.

[21] R.S. Dembo, S.C. Eisenstat and T. Steihaug, Inexact Newton methods, *SIAM J. Numer. Anal.*, **19** (1982), 400-408.

[22] K. Eriksson and C. Johnson, Adaptive streamline diffusion finite element methods for stationary convection-diffusion problems, *Math. Comput.*, **60** (1993), 167-188.

[23] X.P. Guo, On semilocal convergence of inexact Newton methods, *J. Comput. Math.*, **25** (2007), 231-242.

[24] I.E. Kaporin and O. Axelsson, On a class of nonlinear equation solvers based on the residual norm reduction over a sequence of affine subspaces, *SIAM J. Sci. Comput.*, **16** (1995), 228-249.

[25] C.T. Kelley, Iterative Methods for Linear and Nonlinear Equations, *SIAM*, Philadelphia, 1995.

[26] J.M. Ortega and W.C. Rheinboldt, Iterative Solution of Nonlinear Equations in Several Variables, *Academic Press*, New York, 1970.

[27] W.C. Rheinboldt, Methods of Solving Systems of Nonlinear Equations, The Second Edition, *SIAM*, Philadelphia, 1998.

[28] Y. Saad, Iterative Methods for Sparse Linear Systems, The Second Edition, *SIAM*, Philadelphia, 2003.

[29] A.H. Sherman, On Newton-iterative methods for the solution of systems of nonlinear equations, *SIAM J. Numer. Anal.*, **15** (1978), 755-771.