CORE
Metadata, citation and similar papers at core.ac.uk
Provided by Elsevier - Publisher Connector

ELSEVIER

Available online at www.sciencedirect.com
ScienceDirect

Journal of Computational and Applied Mathematics 200 (2007) 47–60

JOURNAL OF COMPUTATIONAL AND APPLIED MATHEMATICS

www.elsevier.com/locate/cam

# A choice of forcing terms in inexact Newton method ☆

Heng-Bin An[a,*], Ze-Yao Mo[b], Xing-Ping Liu[a]

[a] *State Key Laboratory of Computational Physics, Institute of Applied Physics and Computational Mathematics, Beijing, P.O. Box 8009, 100088, PR China*

[b] *High Performance Computing Center, Institute of Applied Physics and Computational Mathematics, Beijing, P.O. Box 8009, 100088, PR China*

Received 13 June 2005; received in revised form 15 December 2005

**Abstract**

Inexact Newton method is one of the effective tools for solving systems of nonlinear equations. In each iteration step of the method, a forcing term, which is used to control the accuracy when solving the Newton equations, is required. The choice of the forcing terms is of great importance due to their strong influence on the behavior of the inexact Newton method, including its convergence, efficiency, and even robustness. To improve the efficiency and robustness of the inexact Newton method, a new strategy to determine the forcing terms is given in this paper. With the new forcing terms, the inexact Newton method is locally $Q$-superlinearly convergent. Numerical results are presented to support the effectiveness of the new forcing terms.
© 2006 Elsevier B.V. All rights reserved.

*MSC:* 65H10; 65F10

*Keywords:* Forcing terms; Inexact Newton method; Newton method; GMRES method; Convergence

## 1. Introduction

In scientific and engineering computing areas, it is often needed to solve the large sparse systems of nonlinear equations

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

where $F : R^n \to R^n$ is a continuously differentiable nonlinear mapping.

Among all kinds of methods for solving the nonlinear equations (1), *Newton method* is one of the most elementary, popular and important one [17]. One of the advantages of the method is its local quadratic convergence. However, its computational cost is expensive, particularly when the size of the problem is very large, because in each iteration step, the *Newton equations*

$$F(x_k) + F'(x_k)s = 0 \tag{2}$$

☆ This project was supported by China Postdoctoral Science Foundation (No. 2005037005), Chinese NSF for DYS (No. 60425205), National Basic Key Research Special Fund (2005CB321702), Chinese NSF (No. 60533020) and Chinese NSF (No. 10571017).
* Corresponding author. Tel.: +86 10 62014411 3191.
*E-mail addresses:* anhengbin@sina.com (H.-B. An), zeyao_mo@iapcm.ac.cn (Z.-Y. Mo), lxp@iapcm.ac.cn (X.-P. Liu).

0377-0427/$ - see front matter © 2006 Elsevier B.V. All rights reserved.
doi:10.1016/j.cam.2005.12.030

should be solved. Here $x_k$ is the current iterate, and $F'(x_k)$ is the Jacobian matrix of $F(x)$ at $x_k$. The solution $s_k^{\mathrm{N}}$ of the Newton equations is the *Newton step*. Once the Newton step is obtained, then the next iterate is given by $x_{k+1} = x_k + s_k^{\mathrm{N}}$.

To reduce the computational cost of Newton method, Dembo, Eisenstat and Steihaug proposed inexact Newton method [7], which is a generalization of Newton method and can be described concisely as follows.

**Algorithm 1.1.** IN (Inexact Newton Method [7])

1. Given $x_0 \in R^n$.
2. For $k = 0, 1, 2, \ldots$ until $\{x_k\}$ convergence
   2.1. Choose **some** $\bar{\eta}_k \in [0, 1)$.
   2.2. Inexactly solve the Newton equations (2) and obtain a step $\bar{s}_k$, such that

$$\|F(x_k) + F'(x_k)\bar{s}_k\| \leqslant \bar{\eta}_k \|F(x_k)\|. \tag{3}$$

   2.3. Let $x_{k+1} := x_k + \bar{s}_k$.

In the above algorithm, $\bar{\eta}_k$ is the *forcing term* for the $k$-th iteration step, $\bar{s}_k$ is the *inexact Newton step* and (3) is the *inexact Newton condition*.

In each iteration step of the inexact Newton method, a real number $\bar{\eta}_k$ should be chosen first, and then an inexact Newton step $\bar{s}_k$ is obtained by solving the Newton equations approximately with an efficient iteration solver for systems of linear equations, such as the classical splitting methods or the modern Krylov subspace methods [20]. Since $F(x_k) + F'(x_k)\bar{s}_k$ is both the residual of the Newton equations and the local linear model of $F(x)$ at $x_k$, the inexact Newton condition (3) reflects essentially both the reduction in the norm of the local linear model and certain accuracy in solving the Newton equations. Thus the role of forcing terms is to control the degree of accuracy when solving the Newton equations. In particular, if $\bar{\eta}_k = 0$ for all $k$, then inexact Newton method will be reduced into Newton method.

Inexact Newton method has been used popularly in many areas, and now it is widely considered that various forms of inexact Newton methods are among the most effective tools for solving systems of nonlinear equations [19]. In particular, if the Krylov subspace iteration methods are used to compute an inexact Newton step, then it leads to a kind of special inexact Newton method, named as *Newton–Krylov* subspace method, which is currently very popular in many application areas [1–4,6,14].

Inexact Newton method, like Newton method, is locally convergent.

**Theorem 1.1** (*Dembo et al. [7, Theorem 2.3]*). *Assume that $F : R^n \to R^n$ is continuously differentiable, $x^* \in R^n$ such that $F(x^*) = 0$ and $F'(x^*)$ is nonsingular. Let $0 < \eta_{\max} < t < 1$ be the given constants. If the forcing terms $\{\bar{\eta}_k\}$ in inexact Newton method satisfy $\bar{\eta}_k \leqslant \eta_{\max} < t < 1$ for all $k$, then there exists $\varepsilon > 0$, such that for any $x_0 \in N_\varepsilon(x^*) \equiv \{x : \|x - x^*\| < \varepsilon\}$, the sequence $\{x_k\}$ generated by inexact Newton method converges to $x^*$, and*

$$\|x_{k+1} - x^*\|_* \leqslant t\|x_k - x^*\|_*,$$

*where $\|y\|_* = \|F'(x^*)y\|$.*

By Theorem 1.1, if the forcing terms $\{\bar{\eta}_k\}$ in inexact Newton method are uniformly strict less than 1, then the method is locally convergent. About the convergence rate of inexact Newton method, we have the following result.

**Theorem 1.2** (*Dembo et al. [7, Corollary 3.5]*). *Assume that $F : R^n \to R^n$ is continuously differentiable, $x^* \in R^n$ such that $F(x^*) = 0$ and $F'(x^*)$ is nonsingular. If the sequence $\{x_k\}$ generated by inexact Newton method converges to $x^*$, then*

(1) *$\{x_k\}$ converges to $x^*$ superlinearly when $\bar{\eta}_k \to 0$;*
(2) *$\{x_k\}$ converges to $x^*$ quadratically if $\bar{\eta}_k = \mathcal{O}(\|F(x_k)\|)$ and $F'(x)$ is Lipschitz continuous at $x^*$.*

Theorem 1.2 shows that the convergence rate of inexact Newton method is determined by the choice of the forcing terms.

When inexact Newton method is used in practical computations, it is necessary to give some way to determine the forcing terms. The choice of the forcing terms is very important for inexact Newton method. It not only determines the asymptotic speed of convergence to a solution of the nonlinear equations (1), but also affects the efficiency and robustness of the algorithm. In fact, if $x_k$ is away from the solution set of the nonlinear equations and a very small $\bar{\eta}_k$ is used to control the accuracy for solving the Newton equations, then it is probable that the obtained step $s_k$ is so bad that $F(x_k+s_k)$ significantly disagrees with the local linear model $F(x_k)+F'(x_k)s_k$. Consequently, "oversolving" phenomenon may occur. This means reducing the linear residual norm without achieving a commensurate reduction in the nonlinear residual norm [11,18,22,23]. Reducing the linear residual norm is not our ultimate purpose; what we really want is to reduce the nonlinear residual norm in each iteration step. "Oversolving" phenomenon introduces much unnecessary computation, or even worse, the whole iteration may be broken down [22,23].

Usually, it is hard to choose a good sequence of forcing terms. In practical computations, many researchers have proposed some concrete strategies. Here we list several representatives.

1. The choice $\bar{\eta}_k = 10^{-4}$ of Cai et al. [5].
2. The choice $\bar{\eta}_k = \min\{1/(k+2), \|F(x_k)\|\}$ of Dembo and Steihaug [8].
3. Eisenstat and Walker [11] proposed two strategies for choosing $\bar{\eta}_k$:
   (a) given $\bar{\eta}_0 \in [0, 1)$, choose

$$\bar{\eta}_k = \frac{\|F(x_k) - F(x_{k-1}) - F'(x_{k-1})s_{k-1}\|}{\|F(x_{k-1})\|}, \quad k = 1, 2, \ldots,$$

   or

$$\bar{\eta}_k = \frac{|\|F(x_k)\| - \|F(x_{k-1}) + F'(x_{k-1})s_{k-1}\||}{\|F(x_{k-1})\|}, \quad k = 1, 2, \ldots.$$

   (b) given $\gamma \in (0, 1]$, $\omega \in (1, 2]$, $\bar{\eta}_0 \in [0, 1)$, choose

$$\bar{\eta}_k = \gamma\left(\frac{\|F(x_k)\|}{\|F(x_{k-1})\|}\right)^{\omega}, \quad k = 1, 2, \ldots.$$

Among the above strategies for choosing forcing terms, the first one never uses any information of $F(x)$, while all the rest use some information about $F(x)$.

At present, the two strategies given by Eisenstat and Walker are more popular and have been used widely. Note that choice (a) reflects the agreement between $F(x)$ and its local linear model at the previous step, while choice (b) reflects the reduction rate of $\|F(x)\|$ from $x_{k-1}$ to $x_k$. Under suitable conditions, the authors proved that if the initial iterate $x_0$ is sufficiently close to a solution $x^*$ of the equations, then the inexact Newton methods resulted from choice (a) or (b) are well-defined, and the iterates $\{x_k\}$ converges to $x^*$. For choice (a), the convergence is $Q$-superlinear, two-step $Q$-quadratic and of $R$-order $(1+\sqrt{5})/2$; for choice (b), the convergence is $Q$-order $\omega$ whenever $\gamma < 1$, or $Q$-order $p$ whenever $\gamma = 1$, where $p \in [1, \omega)$ is arbitrary. In addition, choice (a) and (b) are scale independent: they do not change if $F(x)$ is multiplied by a constant.

In practical computations, for the purpose of preventing the forcing terms from becoming too small too quickly, the authors added some safeguards to choice (a) and (b), consequently the following more concrete strategies are obtained.

(a) Given $\bar{\eta}_0 \in [0, 1)$, choose

$$\bar{\eta}_k = \begin{cases} \xi_k, & \bar{\eta}_{k-1}^{(1+\sqrt{5})/2} \leqslant 0.1, \\ \max\{\xi_k,\ \bar{\eta}_{k-1}^{(1+\sqrt{5})/2}\}, & \bar{\eta}_{k-1}^{(1+\sqrt{5})/2} > 0.1, \end{cases}$$

where

$$\xi_k = \frac{\|F(x_k) - F(x_{k-1}) - F'(x_{k-1})s_{k-1}\|}{\|F(x_{k-1})\|}, \quad k = 1, 2, \ldots, \tag{4}$$

or

$$\xi_k = \frac{|\|F(x_k)\| - \|F(x_{k-1}) + F'(x_{k-1})s_{k-1}\||}{\|F(x_{k-1})\|}, \quad k = 1, 2, \ldots . \tag{5}$$

(b) Given $\gamma \in (0, 1]$, $\omega \in (1, 2]$, $\bar{\eta}_0 \in [0, 1)$, choose

$$\bar{\eta}_k = \begin{cases} \xi_k, & \gamma(\bar{\eta}_{k-1})^{\omega} \leqslant 0.1, \\ \max\{\xi_k, \ \gamma(\bar{\eta}_{k-1})^{\omega}\}, & \gamma(\bar{\eta}_{k-1})^{\omega} > 0.1, \end{cases}$$

where

$$\xi_k = \gamma\left(\frac{\|F(x_k)\|}{\|F(x_{k-1})\|}\right)^{\omega}, \quad k = 1, 2, \ldots .$$

The numerical experiments in [11] show that the above two choices can effectively overcome the "oversolving" phenomenon, and thus improve the efficiency of inexact Newton method. Besides, choice (a) and choice (b) with $\gamma \geqslant 0.9$ and $\omega \geqslant (1 + \sqrt{5})/2$ have the best performances.

The choice of the forcing terms should be related to specific problems and the information of $F(x)$ should be used effectively. Note that the choices given by Eisenstat and Walker use some information of $F(x)$, but choice (a) only reflects the agreement between $F(x)$ and its local linear model, while choice (b) only reflects the rate of reduction of $\|F(x)\|$. Is it possible to give a choice that can reflect both of the two aspects?

We propose a new way of choosing forcing terms that can reflect not only the agreement between $F(x)$ and its local linear model, but also the rate of reduction of $\|F(x)\|$ in some sense. In addition, like choices 3, our choice is also scale independent. Numerical results show that the new choice is more effective than all the above strategies 1–3.

In Section 2, we present the new strategy for choosing forcing terms and analyze the local convergence of the corresponding inexact Newton method. In Section 3, some numerical results are given to show the efficiency of our strategy; and finally, we give a short conclusion in Section 4.

## 2. The new choice

Assume that $x_k$ is the current iterate, $s_k$ is a step from $x_k$. The *actual reduction* $\text{Ared}_k(s_k)$ and *predicted reduction* $\text{Pred}_k(s_k)$ of $F(x)$ at $x_k$ with step $s_k$ are defined, respectively, as

$$\text{Ared}_k(s_k) = \|F(x_k)\| - \|F(x_k + s_k)\|,$$

$$\text{Pred}_k(s_k) = \|F(x_k)\| - \|F(x_k) + F'(x_k)s_k\|.$$

Furthermore, let

$$r_k = \frac{\text{Ared}_k(s_k)}{\text{Pred}_k(s_k)}.$$

In trust region method, $r_k$ is used to adjust the radii of the trust regions [9,15]. Here we use $r_k$ to adjust the forcing term $\bar{\eta}_k$.

Assume that $F(x_k) \neq 0$. By inexact Newton condition (3), we know that $\text{Pred}_k(s_k) \geqslant (1 - \bar{\eta}_k)\|F(x_k)\| > 0$. Therefore, if $r_k \approx 1$, then the local linear model and nonlinear model will agree well on their scale, and at this time, $\|F(x)\|$ usually will be reduced obviously; if $r_k$ nears 0 but $r_k > 0$, then the local linear model and nonlinear model disagree and $\|F(x)\|$ will be reduced very little; if $r_k < 0$, then the local linear model and nonlinear model disagree and $\|F(x)\|$ will be enlarged; finally, if $r_k \gg 1$, then the local linear model and nonlinear model also disagree, but at this time, $\|F(x)\|$ will be reduced greatly.

Usually, we hope that the local linear model and nonlinear model agree well, thus the case $r_k \approx 1$ is the best one because in this situation, the linear model and nonlinear model agree at least on scale; besides, the case $r_k \gg 1$ is relatively acceptable because it leads to a great reduction point; however, the worst case is that $r_k$ nears or is less than 0, because at this point we cannot obtain anything useful. According to the above property of $r_k$, we can choose forcing terms by the following way:

$$\bar{\eta}_k = \begin{cases} 1-2p_1, & r_{k-1} < p_1, \\ \bar{\eta}_{k-1}, & p_1 \leqslant r_{k-1} < p_2, \\ 0.8\bar{\eta}_{k-1}, & p_2 \leqslant r_{k-1} < p_3, \\ 0.5\bar{\eta}_{k-1}, & r_{k-1} \geqslant p_3, \end{cases} \quad k = 1, 2, \ldots, \tag{6}$$

where $0 < p_1 < p_2 < p_3 < 1$ are prescribed at first, and $p_1 \in (0, \frac{1}{2})$. In addition, assume that $\bar{\eta}_0$ is given.

Our choice of forcing terms is to determine $\bar{\eta}_k$ by the magnitude of $r_{k-1}$. If $r_{k-1} < p_1$, i.e., $r_{k-1}$ is relatively small, then it is possible that the property of $F(x)$ at the iterate $x_k$ is not so good such that $F(x)$ and its local linear model cannot agree well. In this case, let $\bar{\eta}_k = 1 - 2p_1$ ($p_1$ is small, and so $1 - 2p_1$ is relatively large), and relax the accuracy for solving the Newton equations. If $r_{k-1}$ is relatively large ($r_{k-1} \geqslant p_2$), then $F(x)$ and its linear model agree well (the case $r_{k-1} \geqslant 1$ is quite few in our experiments, so we omit this case), and $\|F(x)\|$ will be reduced obviously. In this case, shrink $\bar{\eta}_k$ properly so that the Newton equations can be solved more accurately. Otherwise ($p_1 \leqslant r_{k-1} < p_2$), $\bar{\eta}_k$ is not changed.

It should be noted that the current forcing term $\bar{\eta}_k$ is determined by the previous value $r_{k-1}$; while $\bar{\eta}_k$ determines the current value $r_k$ through solving the Newton equations approximately. Thus, the sequences $\{\bar{\eta}_k\}$ and $\{r_k\}$ are interrelated. In practical computations, there is no warranty that the current forcing term $\bar{\eta}_k$ is always proper (that is, with this $\bar{\eta}_k$, the local linear model and nonlinear model can agree relatively well.). In particular, something unexpected may happen: in some sequent nonlinear iterations, if some relatively small $\bar{\eta}_k$ is used, then the corresponding $r_k$ may be large; but the actual values of $\bar{\eta}_k$ are so large that $r_k < p_1$. In order to prevent the happening of such unexpected occasions, we modify the new choice as follows:

$$\bar{\eta}_k \longleftarrow 0.5\bar{\eta}_{k-1} \quad \text{whenever } \bar{\eta}_{k-2}, \bar{\eta}_{k-1} > 0.1 \quad \text{and} \quad r_{k-2}, \quad r_{k-1} < p_1. \tag{7}$$

In the modification above, the value $\bar{\eta}_{k+1}$ is set to be half of the previous forcing term only if, in the previous sequent two steps, the forcing terms are larger than a threshold 0.1 and the ratios of actual reduction to predicted reduction are less than $p_1$.

**Remark 2.1.** The shrinking factors 0.8 and 0.5 in (6) and the threshold value 0.1 in (7) are arbitrary. But these values are more effective in our experiments.

**Remark 2.2.** The new choice of $\{\bar{\eta}_k\}$ can be refined further. For example, we may give much more $p_j$ and determine $\bar{\eta}_k$ in more different cases according to the magnitude of $r_k \to r_{k-1}$.

To obtain the local convergence of the inexact Newton method with the new forcing terms, we need the following lemmas.

**Lemma 2.1** (*Ortega and Rheinboldt [17, 2.3.3]*). *Assume that $F : R^n \to R^n$ is continuously differentiable and $x \in R^n$. If $F'(x)$ is nonsingular, then for any $\varepsilon > 0$, there exists $\delta > 0$, such that $F'(y)$ is nonsingular and*

$$\|F'(y)^{-1} - F'(x)^{-1}\| < \varepsilon$$

*whenever $y \in N_\delta(x)$.*

**Lemma 2.2** (*Ortega and Rheinboldt [17, 3.2.10]*). *Assume that $F : R^n \to R^n$ is continuously differentiable. Then for any $z \in R^n$ and $\varepsilon > 0$, there exists $\delta > 0$, such that*

$$\|F(x) - F(y) - F'(y)(x - y)\| < \varepsilon\|x - y\|$$

*whenever $x, y \in N_\delta(z)$.*

Now we can show that the inexact Newton method with our forcing terms is locally convergent.

**Theorem 2.1.** *Assume that $F : R^n \to R^n$ is continuously differentiable, $x^* \in R^n$ such that $F(x^*) = 0$ and $F'(x^*)$ is nonsingular. Given $\bar{\eta}_0 \in (0, 1)$, $0 < p_1 < p_2 < p_3 < 1$, and let $p_1 \in (0, \frac{1}{2})$. If $x_0$ is sufficiently close to $x^*$, then the sequence $\{x_k\}$ produced by the inexact Newton method with forcing terms* (6) *and* (7) *converges to $x^*$ Q-superlinearly.*

**Proof.** Let $\hat{\eta} = \max\{\bar{\eta}_0, 1 - 2p_1\}$ and $\beta = \|F'(x^*)^{-1}\|$, then it is easy to see that $\hat{\eta} \in (0, 1)$. By Lemmas 2.1 and 2.2, there exists $\delta > 0$, such that $F'(x)^{-1}$ is invertible and the inequalities

$$\|F'(x)^{-1}\| < 2\beta \tag{8}$$

and

$$\|F(x) - F(y) - F'(y)(x - y)\| < \frac{(1 - p_3)(1 - \hat{\eta})}{2\beta(1 + \hat{\eta})}\|x - y\| \tag{9}$$

hold whenever $x, y \in N_\delta(x^*)$.

By (6) and (7), it is obvious that

$$\bar{\eta}_k \leqslant \hat{\eta} < 1, \quad k = 1, 2, \ldots,$$

thus by Theorem 1.1, the sequence $\{x_k\}$ converges to $x^*$. This shows that there exists a positive integer $K$, such that $x_k \in N_\delta(x^*)$ whenever $k > K$. Therefore, for all $k > K$, by the inexact Newton condition and (8), we have

$$\|s_k\| = \|F'(x_k)^{-1}[F'(x_k)s_k + F(x_k) - F(x_k)]\| \leqslant 2\beta(1 + \hat{\eta})\|F(x_k)\|.$$

Consequently, for all $k > K$, the inexact Newton condition and (9) show that

$$\begin{aligned}
r_k &= \frac{\|F(x_k)\| - \|F(x_k + s_k)\|}{\|F(x_k)\| - \|F(x_k) + F'(x_k)s_k\|} \\
&\geqslant \frac{\|F(x_k)\| - \|F(x_k + s_k) - F(x_k) - F'(x_k)s_k\| - \|F(x_k) + F'(x_k)s_k\|}{\|F(x_k)\| - \|F(x_k) + F'(x_k)s_k\|} \\
&= 1 - \frac{\|F(x_k + s_k) - F(x_k) - F'(x_k)s_k\|}{\|F(x_k)\| - \|F(x_k) + F'(x_k)s_k\|} \\
&\geqslant 1 - \frac{\dfrac{(1 - p_3)(1 - \hat{\eta})}{2\beta(1 + \hat{\eta})}\|s_k\|}{(1 - \bar{\eta}_k)\|F(x_k)\|} \\
&\geqslant 1 - \frac{(1 - p_3) \cdot 2\beta(1 + \hat{\eta})\|F(x_k)\|}{2\beta(1 + \hat{\eta})\|F(x_k)\|} \\
&= p_3.
\end{aligned}$$

Thus, (6) shows that $\bar{\eta}_k \to 0$. So by Theorem 1.2, $\{x_k\}$ converges to $x^*$ $Q$-superlinearly. □

From the proof of the theorem, we know that $r_k > p_3$ for all sufficiently large $k$. Thus, by (6), the forcing terms are shrank by half for all $k$ large enough.

## 3. Numerical results

In this section, we present three numerical examples to show the efficiency of the new strategy to choose forcing terms. We compare the new strategy with some old strategies on their numerical behavior.

### 3.1. The algorithm

Since the initial guesses for inexact Newton method cannot always be guaranteed to be near a solution of the nonlinear systems, we use an inexact Newton method globalized by backtracking strategy [10], which can be described as follows:

**Algorithm 3.1.** INB (Inexact Newton Backtracking Method [10])

1. Given $x_0 \in R^n$, $\eta_{\max} \in [0, 1)$, $\alpha \in (0, 1)$ and $0 < \theta_{\min} < \theta_{\max} < 1$.
2. For $k = 0, 1, 2, \dots$ until $\{x_k\}$ convergence
   2.1. Choose **some** $\bar{\eta}_k \in [0, \eta_{\max}]$.
   2.2. Inexactly solve the Newton equations (2) and obtain a step $\bar{s}_k$, such that

$$\|F(x_k) + F'(x_k)\bar{s}_k\| \leqslant \bar{\eta}_k \|F(x_k)\|.$$

   2.3. Implement backtracking loop:
   2.3.1 Let $s_k = \bar{s}_k$, $\eta_k = \bar{\eta}_k$.
   2.3.2 While $\|F(x_k + s_k)\| > [1 - \alpha(1 - \eta_k)]\|F(x_k)\|$
   - Choose $\theta \in [\theta_{\min}, \theta_{\max}]$.
   - Update $s_k \longleftarrow \theta s_k$ and $\eta_k \longleftarrow 1 - \theta(1 - \eta_k)$.

2.4. Let $x_{k+1} = x_k + s_k$.

In each iteration step of the above algorithm, when an inexact Newton step $\bar{s}_k$ at level $\bar{\eta}_k$ is obtained, then the backtracking loop along $\bar{s}_k$ is implemented until the condition

$$\|F(x_k + s_k)\| \leqslant [1 - \alpha(1 - \eta_k)]\|F(x_k)\| \tag{10}$$

is satisfied. (10) is called the *sufficient decrease condition*, and it is used to guarantee that $\|F(x_{k+1})\|$ has a certain decrease in each iteration step [2,10,11]. In theory, Lemma 5.1 in [10] guarantees that the backtracking loop will be terminated in finite iterations. In practical computations, a positive integer usually is given in advance to control the maximal backtracking loop number along $\bar{s}_k$ [2,11,23].

Theorem 6.1 in [10] shows that if a sequence $\{x_k\}$ produced by Algorithm INB has a limit point $x^*$ such that $F'(x^*)$ is invertible, then $F(x^*) = 0$ and $x_k \to x^*$. Furthermore, in this case, $\bar{\eta}_k$ and $\bar{s}_k$ can be accepted for all sufficiently large $k$; in particular, it follows that the ultimate convergence rate of $\{x_k\}$ is determined by the choice of the forcing terms $\{\bar{\eta}_k\}$.

It should be noted that in Algorithm INB, each $\bar{\eta}_k$ is required to be less than $\eta_{\max}$, so additional safeguard

$$\bar{\eta}_k \longleftarrow \min\{\bar{\eta}_k, \eta_{\max}\}$$

is needed for the strategies of Eisenstat and Walker. As for our new strategy, since

$$\bar{\eta}_k \leqslant \hat{\eta} \equiv \max\{\bar{\eta}_0, 1 - 2p_1\}$$

for all $k$ (see the proof of Theorem 2.1), no safeguard is needed.

In our tests, GMRES method [21] without restarting is employed to produce an inexact Newton step $\bar{s}_k$. In addition, in GMRES iterations, the starting vector $s_k^0 = 0$ and the number of maximal iteration steps is 40. This number is large enough because in our test the GMRES iteration never reaches 40.

For any $x,\ y \in R^n$, the product $F'(x)y$ is approximately computed by the finite difference formula

$$F'(x)y \approx \frac{F(x + \varepsilon y) - F(x)}{\varepsilon}, \quad \varepsilon = 10^{-7}\frac{\|x\|}{\|y\|}.$$

Thus Jacobian matrix is never formed. See [2,3,11].

In the while loop of Algorithm INB, each $\theta$ is chosen by minimizing a quadratic polynomial $p(\theta)$ on $[\theta_{\min}, \theta_{\max}]$, where $p(\theta)$ is obtained by interpolation such that $p(0) = g(0)$, $p'(0) = g'(0)$ and $p(1) = g(1)$, with $g(\theta) = \|F(x_k + \theta s_k)\|_2^2$. See [2,11].

The norm in our tests is Euclidean norm $\|\cdot\|_2$ and the stopping criteria is that either the current iterate $x_k$ satisfies

$$\max\left\{\frac{1}{\sqrt{n}}\|F(x_k)\|, \frac{\|F(x_k)\|}{\|F(x_0)\|}\right\} \leqslant 10^{-6}, \tag{11}$$

or the number of iteration steps has exceeded 300. Within each nonlinear iteration, the maximal backtracking number along inexact Newton direction $\bar{s}_k$ is limited by 20. In addition, a failure is declared if any of the following three situations occurs during the iteration process:

$F_1$. The number of nonlinear iteration reaches 300, but no $x_k$ satisfies (11) is obtained;
$F_2$. In one iteration, the backtracking number reaches 20 but no satisfactory step is produced; and
$F_3$. $|\|F(x_{k-1})\| - \|F(x_k)\|| \leqslant 10^{-6}\|F(x_k)\|$, which means that the iteration sequence of Algorithm INB cannot manage to escape from a local minimizer of the function $\|F(x)\|$, see [12,13].

### 3.2. Test problems and results

Our test problems are all typical systems of nonlinear equations in literature, with each of its own name and standard initial guess, say $x_s$. The problems and their standard initial guesses are listed as follows.

**Problem 3.1** (*Generalized function of Rosenbrock [16]*).

$$\begin{cases} f_1(x) = -4c(x_2 - x_1^2)x_1 - 2(1 - x_1), \\ f_i(x) = 2c(x_i - x_{i-1}^2) - 4c(x_{i+1} - x_i^2)x_i - 2(1 - x_i), \quad i = 2, 3, \ldots, n-1, \\ f_n(x) = 2c(x_n - x_{n-1}^2), \quad c = 2, \end{cases}$$

with $x_s = (1.2, 1.2, \ldots, 1.2)^T$. We test the case of $n = 5000$.

**Problem 3.2** (*Tridiagonal system [15]*).

$$\begin{cases} f_1(x) = 4(x_1 - x_2^2), \\ f_i(x) = 8x_i(x_i^2 - x_{i-1}) - 2(1 - x_i) + 4(x_i - x_{i+1}^2), \quad i = 2, 3, \ldots, n-1, \\ f_n(x) = 8x_n(x_n^2 - x_{n-1}) - 2(1 - x_n), \end{cases}$$

with $x_s = (12, 12, \ldots, 12)^{\mathrm{T}}$, we test the case of $n = 6000$.

**Problem 3.3** (*Five-diagonal system [15]*).

$$\begin{cases} f_1(x) = 4(x_1 - x_2^2) + x_2 - x_3^2, \\ f_2(x) = 8x_2(x_2^2 - x_1) - 2(1 - x_2) + 4(x_2 - x_3^2) + x_3 - x_4^2, \\ f_i(x) = 8x_i(x_i^2 - x_{i-1}) - 2(1 - x_i) + 4(x_i - x_{i+1}^2) + x_{i-1}^2 - x_{i-2} + x_{i+1} - x_{i+2}^2, \quad i = 3, 4, \ldots, n-2, \\ f_{n-1}(x) = 8x_{n-1}(x_{n-1}^2 - x_{n-2}) - 2(1 - x_{n-1}) + 4(x_{n-1} - x_n^2) + x_{n-2}^2 - x_{n-3}, \\ f_n(x) = 8x_n(x_n^2 - x_{n-1}) - 2(1 - x_n) + x_{n-1}^2 - x_{n-2}, \end{cases}$$

with $x_s = (-2, -2, \ldots, -2)^{\mathrm{T}}$, we test the case of $n = 5000$.

Besides the standard initial guess $x_s$, we also test other initial guesses such as $x_0 = 0$, $x_0 = \pm jx_s$, $j = 1, 2, \ldots, 5$ and $x_0 = je$, $j = 2, 3, \ldots, 5$, where 0 denotes the zero vector and $e$ represents the vector with all entries being 1. It is easy to see that $e$ is a solution to each of the above three problems. For each problem, certain representative results of the initial guesses are listed in Tables 1–3.

Table 1
Results for Generalized function of Rosenbrock

| Choice | | $x_0$ | | | | | | | | | | AN |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | $x_s$ | $2x_s$ | $3x_s$ | $4x_s$ | $5x_s$ | $2e$ | $3e$ | $4e$ | $5e$ | 0 | |
| New | NI | 6 | 10 | 14 | 12 | 12 | 9 | 10 | 14 | 14 | 8 | 10.9 |
| | GI | 33 | 64 | 69 | 79 | 71 | 57 | 58 | 72 | 60 | 40 | 60.3 |
| | FE | 40 | 78 | 84 | 94 | 85 | 67 | 70 | 89 | 76 | 50 | 73.3 |
| EW1 | NI | 7 | 15 | 26 | 22 | 20 | 14 | 20 | 28 | 21 | 10 | 18.3 |
| | GI | 42 | 63 | 161 | 137 | 91 | 73 | 119 | 188 | 134 | 54 | 106.2 |
| | FE | 50 | 81 | 219 | 184 | 122 | 93 | 156 | 254 | 177 | 66 | 140.2 |
| EW2 | NI | 5 | 9 | 21 | 15 | 32 | 11 | 11 | 24 | 15 | 9 | 15.2 |
| | GI | 37 | 52 | 95 | 66 | 144 | 53 | 46 | 96 | 65 | 52 | 70.6 |
| | FE | 43 | 62 | 129 | 84 | 213 | 65 | 59 | 135 | 81 | 63 | 93.4 |
| CGKT | NI | 4 | 7 | 8 | 9 | 10 | 6 | 8 | 8 | 9 | 7 | 7.6 |
| | GI | 46 | 83 | 78 | 95 | 98 | 69 | 97 | 81 | 95 | 93 | 83.5 |
| | FE | 51 | 91 | 87 | 105 | 109 | 76 | 106 | 90 | 105 | 102 | 92.2 |
| DS | NI | 7 | 11 | 14 | 22 | 36 | 10 | 15 | 24 | 20 | 9 | 16.8 |
| | GI | 36 | 63 | 76 | 149 | 310 | 52 | 91 | 161 | 125 | 48 | 111.1 |
| | FE | 44 | 78 | 98 | 198 | 430 | 64 | 116 | 223 | 166 | 59 | 147.6 |

Table 2
Results for tridiagonal system

| Choice | | $x_0$ | | | | | | | | | | AN |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | $x_s$ | $2x_s$ | $3x_s$ | $4x_s$ | $5x_s$ | $2e$ | $3e$ | $4e$ | $5e$ | 0 | |
| New | NI | 12 | 20 | 15 | 29 | 32 | 9 | 10 | 12 | 10 | 8 | 15.7 |
| | GI | 60 | 184 | 83 | 225 | 239 | 55 | 58 | 78 | 51 | 40 | 107.3 |
| | FE | 74 | 215 | 101 | 279 | 301 | 64 | 70 | 94 | 63 | 50 | 131.1 |
| SW1 | NI | 37 | 127 | * | * | * | 14 | 17 | 28 | 22 | 10 | 36.4 |
| | GI | 264 | 1694 | * | * | * | 69 | 95 | 188 | 121 | 52 | 354.7 |
| | FE | 357 | 2430 | * | * | * | 87 | 126 | 254 | 160 | 64 | 496.9 |
| SW2 | NI | 70 | 218 | * | * | * | 11 | 11 | 23 | 13 | 9 | 50.7 |
| | GI | 349 | 1411 | * | * | * | 56 | 58 | 101 | 66 | 52 | 299.0 |
| | FE | 616 | 2678 | * | * | * | 68 | 71 | 137 | 80 | 63 | 530.4 |
| CGKT | NI | 11 | 42 | 43 | 37 | 30 | 6 | 8 | 9 | 9 | 8 | 20.3 |
| | GI | 95 | 580 | 582 | 466 | 333 | 61 | 84 | 97 | 78 | 153 | 252.9 |
| | FE | 107 | 692 | 695 | 546 | 378 | 68 | 93 | 107 | 88 | 175 | 294.9 |
| DS | NI | 32 | 212 | * | * | * | 10 | 15 | 24 | 19 | 9 | 45.9 |
| | GI | 240 | 2715 | * | * | * | 52 | 91 | 161 | 114 | 48 | 488.7 |
| | FE | 324 | 4258 | * | * | * | 64 | 116 | 223 | 150 | 59 | 742.0 |

For the convenience of reporting the results of different forcing terms, the following notations are used.

- New: new strategy;
- EW1: the first strategy given by Eisenstat and Walker [11];
- EW2: the second strategy given by Eisenstat and Walker [11];
- CGKT: the strategy used by Cai et al. [5];
- DS: the strategy of Dembo and Steihaug [8].

Table 3
Results for five-diagonal system

| Choice | | $x_0$ | | | | | | | | | | AN |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | $-x_s$ | $-2x_s$ | $-3x_s$ | $-4x_s$ | $-5x_s$ | $2e$ | $3e$ | $4e$ | $5e$ | 0 | |
| New | NI | 8 | 11 | 11 | 17 | 12 | 8 | 10 | 11 | 14 | 9 | 11.1 |
| | GI | 40 | 65 | 55 | 80 | 58 | 40 | 56 | 65 | 57 | 46 | 56.2 |
| | FE | 49 | 77 | 68 | 103 | 72 | 49 | 68 | 77 | 73 | 57 | 69.3 |
| SW1 | NI | 10 | 15 | 21 | 27 | 73 | 10 | 18 | 15 | 22 | 11 | 22.2 |
| | GI | 50 | 57 | 97 | 177 | 759 | 50 | 94 | 57 | 131 | 54 | 152.6 |
| | FE | 61 | 75 | 132 | 241 | 1061 | 61 | 126 | 75 | 175 | 67 | 207.4 |
| SW2 | NI | 11 | 21 | 27 | 38 | 55 | 11 | 11 | 21 | 15 | 9 | 21.9 |
| | GI | 42 | 90 | 109 | 158 | 210 | 42 | 45 | 90 | 53 | 41 | 88.0 |
| | FE | 54 | 123 | 156 | 249 | 390 | 54 | 58 | 123 | 71 | 52 | 133.0 |
| CGKT | NI | 7 | 13 | 14 | 10 | 11 | 7 | 8 | 13 | 10 | 7 | 10.0 |
| | GI | 83 | 176 | 169 | 94 | 98 | 83 | 83 | 176 | 106 | 104 | 117.2 |
| | FE | 91 | 198 | 188 | 105 | 110 | 91 | 92 | 198 | 117 | 113 | 130.3 |
| DS | NI | 11 | 21 | 20 | 26 | 67 | 11 | 16 | 21 | 20 | 9 | 22.2 |
| | GI | 58 | 123 | 112 | 168 | 620 | 58 | 89 | 123 | 117 | 43 | 151.1 |
| | FE | 73 | 172 | 153 | 229 | 926 | 73 | 119 | 172 | 157 | 54 | 212.8 |

In our implementing of Algorithm INB, the parameters $\alpha = 0.5$, $\theta_{\min} = 0.1$ and $\theta_{\max} = 0.5$ are used. In addition, $\eta_{\max} = 0.9$ is used for EW1 and EW2 forcing terms and the initial forcing term $\bar{\eta}_0 = 0.5$ is used for New, EW1 and EW2 strategies. We always use $\bar{\eta}_k$ given by (5) for EW1 since this expression is more convenient to be used; for EW2, we use $\gamma = 0.9$ and $\omega = 2$; for the new strategy, $p_1 = 0.1$, $p_2 = 0.4$ and $p_3 = 0.7$ are used.

In the reporting tables, the following notations are used.

- NI: represents total number of nonlinear iterations;
- GI: represents total number of GMRES iterations;
- FE: denotes the total function evaluation number;
- AN: denotes the average number for NI, GI and FE;
- BT: represents the backtracking number along the inexact Newton direction; and
- $*$: denotes a failure occurred in either situations $F_1$–$F_3$.

For each of Problems 3.1–3.3, the results of ten initial guesses are separately given in Tables 1–3, where the numbers of NI, GI and FE are listed and their averages for ten cases are given in the last columns.

Table 1 is the result of Problem 3.1 for ten initial guesses. From the table, one sees that the average nonlinear iteration number of Algorithm INB with New forcing term is 10.9, which is a little more than 7.6, the average nonlinear iteration number of Algorithm INB with CGKT forcing term; while the least average number of NI for EW1, EW2 and DS forcing terms is 15.4. If comparing the average numbers of GMRES iteration and function evaluation, we see clearly that New forcing term is the winner. In particular, Algorithm INB with New forcing term needs averagely 73.3 function evaluations while the algorithm with other forcing terms needs at least 92.2 function evaluations. This shows that New forcing term is the most efficient one for this problem.

In Table 2, we give the result about ten initial guesses for Problem 3.2. First we see that when $x_0 = 3x_s$, $4x_x$ and $5x_s$, Algorithm INB with EW1, EW2 and DS forcing terms cannot solve Problem 3.2 successfully, that is, for each case, one failed situation among $F_1$–$F_3$ occurs. Note that the last column of average numbers for NI, GI and FE only include the successful cases. From this table, one sees that the average numbers of NI, GI and FE for New forcing terms are the least. Furthermore, it is easy to see that the numbers of FE for New forcing term are the least for all ten cases. The average function evaluation number of New forcing term is 131.1, while the minimal average function evaluations of all the other four forcing terms is 294.9, at least twice over the former number, and the maximum is 742.0, about five times over it. Therefore, New forcing term is the most effective strategy for Problem 3.2.

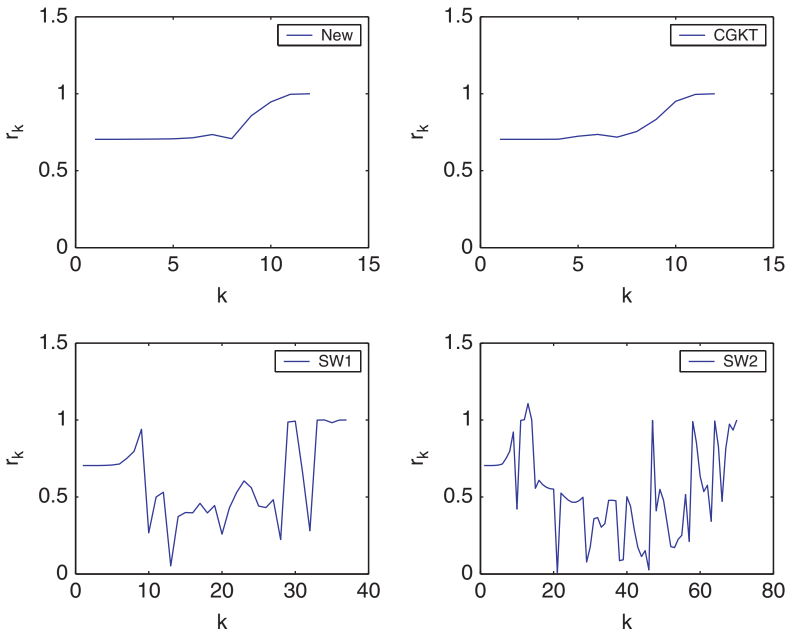


Fig. 1. The curves of $r_k$ for different forcing terms in Problem 3.2 with $x_0 = x_s$.

Table 3 is the result of Problem 3.3. From this table, one sees that averagely, the number of NI for New is 11.1, which is a little more than 10.0, the average number of NI for CGKT; while the rest three forcing terms need at least 21.9 nonlinear iterations on average. If comparing the average GMRES iteration numbers and function evaluation numbers, then we can obtain the conclusion that New forcing term gives the best performance. The average function evaluation number of New forcing term is 69.3, while those of all others exceed 100. Thus, Algorithm INB with New forcing term performs most effectively to solve Problem 3.3.

By the above observation and analysis for Tables 1–3, we see that the average numbers of function evaluation of Algorithm INB with New forcing terms are the least, thus the new strategy is the most effective one to choose forcing terms in inexact Newton method. Besides, it is easy to see that DS forcing terms need the most function evaluations on average. Thus, this strategy is the worst in our tests. In the following, we do not concern this strategy any more.

Since the principle of inexact Newton or Newton method is to use the local linear model to replace the nonlinear model, we expect that the local linear model and nonlinear model can agree as well as possible. To compare this aspect of different forcing terms, we depict three figures about Problem 3.2 when $x_0 = x_s$, $2x_s$ and $3x_s$. In these figures, we do not consider the DS forcing terms because, as discussed above, their efficiency is the worst.

In Fig. 1, when $x_0 = x_s$ in Problem 3.2, the curves of $r_k$ under four forcing terms are plotted, where $k$ denotes the iteration index. This figure shows that $r_k$ of New and CGKT are always larger than 0.5, and their curves are relatively stable with iteration index; while the curves of SW1 and SW2 are unstable, with the curve of SW2 is the worst. As is discussed in Section 2, the more $r_k$ nears 1, the better the local linear model agrees the nonlinear model. Therefore, Fig. 1 shows that the local linear model and nonlinear model under New or CGKT forcing terms can agree relatively well. Consequently, numerical results of New and CGKT forcing terms are better than those of EW1 and EW2. For the curve of EW2, we see that $r_k$ is almost zero when $k = 21$. This shows that $F(x_k) + F'(x_k)s_k$ and $F(x_k + s_k)$ disagree significantly for $k = 21$. At the same time, $\|F(x_k + s_k)\|$ is only little reduced when compared to $\|F(x_k)\|$ for $k = 21$. This kind of case is the worst.

Fig. 2 is of the curves of $r_k$ for different forcing terms when $x_0 = 2x_s$ in Problem 3.2. From this figure, we see that except $k = 14$ and 16, the values of $r_k$ for New forcing term are always larger than 0.5, and the curve of $r_k$ for New forcing term is relatively stable compared to those of others. This shows that under New forcing terms, local linear model can represent nonlinear model for almost all $k$, thus it is the best one. The better one is CGKT since its curve is a little worse than New forcing term's and better than others'. The curves of EW1 and EW2 are the worst because they

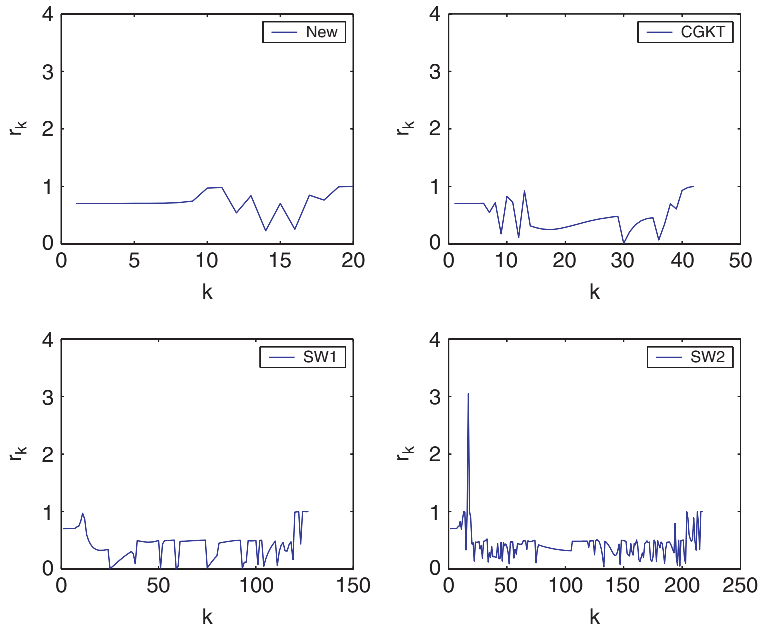


Fig. 2. The curves of $r_k$ for different forcing terms in Problem 3.2 with $x_0 = 2x_s$.

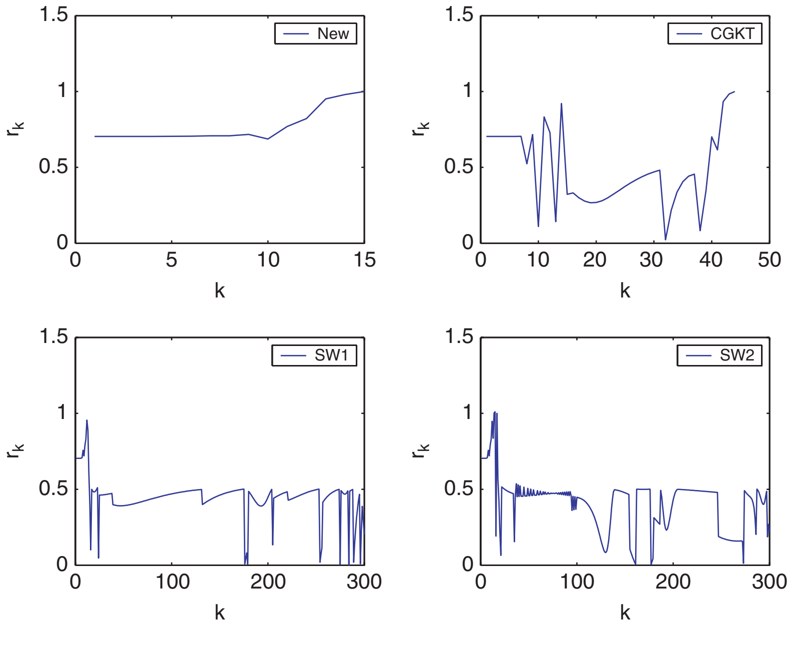


Fig. 3. The curves of $r_k$ for different forcing terms in Problem 3.2 with $x_0 = 3x_s$.

are very unstable and relatively low. This shows that the local linear model disagrees nonlinear model in most cases under SW1 and SW2 forcing terms, thus $\|F(x_k)\|$ cannot be decreased effectively. We see that there is a big jump that reaches 3 on the curve of SW2, this shows that the local linear model and nonlinear model disagree considerably for some $k$, but fortunately, $\|F(x_{k+1})\|$ is significantly smaller than $\|F(x_k)\|$.

Table 4
Iteration comparison between CGKT and New forcing terms on tridiagonal system with $x_0 = x_s$

| $k$ | CGKT | | | | New | | | | | $\|x_k^{\text{CGKT}} - x_k^{\text{New}}\|$ |
|---|---|---|---|---|---|---|---|---|---|---|
| | GI | $\|F(x_k)\|$ | $r_k$ | BT | GI | $\|F(x_k)\|$ | $r_k$ | BT | $\bar{\eta}_k$ | |
| 1 | 5 | 2.792e+5 | 0.704 | 0 | 1 | 2.792e+5 | 0.704 | 0 | 0.2500 | 1.303e+2 |
| 2 | 3 | 8.269e+4 | 0.704 | 0 | 1 | 8.270e+4 | 0.704 | 0 | 0.1250 | 1.302e+2 |
| 3 | 3 | 2.448e+4 | 0.704 | 0 | 1 | 2.448e+4 | 0.704 | 0 | 0.0625 | 1.301e+2 |
| 4 | 4 | 7.233e+3 | 0.707 | 0 | 1 | 7.234e+3 | 0.705 | 0 | 0.0313 | 1.301e+2 |
| 5 | 11 | 4.615e+3 | 0.724 | 1 | 1 | 2.123e+3 | 0.707 | 0 | 0.0156 | 8.441e+1 |
| 6 | 8 | 1.219e+3 | 0.736 | 0 | 1 | 6.097e+2 | 0.714 | 0 | 0.0078 | 1.907e+1 |
| 7 | 10 | 3.435e+2 | 0.718 | 0 | 2 | 1.625e+2 | 0.735 | 0 | 0.0039 | 1.695e+1 |
| 8 | 10 | 8.456e+1 | 0.754 | 0 | 10 | 1.050e+2 | 0.708 | 1 | 0.0020 | 8.198e+0 |
| 9 | 12 | 1.410e+1 | 0.833 | 0 | 8 | 1.520e+1 | 0.857 | 0 | 0.0010 | 2.507e−1 |
| 10 | 14 | 6.907e−1 | 0.951 | 0 | 10 | 8.152e−1 | 0.947 | 0 | 0.0005 | 4.697e−2 |
| 11 | 13 | 3.204e−3 | 0.995 | 0 | 11 | 2.840e−3 | 0.997 | 0 | 0.0002 | 2.287e−4 |

When $x_0 = 3x_s$ for Problem 3.2, the curves of $r_k$ for different forcing terms are plotted in Fig. 3. From this figure, we see that the curve of New forcing terms is obviously the best one, because only it lies above 0.5 for all $k$ and stably approximates 1 with the increase of $k$. This shows that under New forcing terms, the local linear model and nonlinear model agree well, thus the nonlinear model can be decreased significantly for each $k$. The curve of CGKT can be considered the better one. The worst are the curves of SW1 and SW2. These two curves are unstable and most part lie under 0.5. This shows that the local linear model and nonlinear model disagree considerably and $\|F(x_k)\|$ can only be decreased very little in each iteration. As a result, after 300 nonlinear iterations, $x_k$ is still dissatisfied. See Table 2.

From Figs. 1–3, we see that, for all convergent cases of each forcing terms, $r_k$ is almost 1 in the last several iteration steps. This means that the local linear model and nonlinear model under different forcing terms agree very well when $x_k$ nears the solution of the nonlinear equations.

The above comparison about $r_k$ for different forcing terms shows that New forcing term is the most effective one. Under this forcing term, the local linear model and nonlinear model can agree well, so the nonlinear residual can be decreased effectively by solving the Newton equations to proper degree.

One sees from Fig. 1 that the curves of $r_k$ for New and CGKT differ obscurely when $x_0 = x_s$ in Problem 3.2, but Table 2 shows that the GMRES iterations and function evaluations under these two forcing terms differ significantly. To see the variation of some values in the iteration process, see Table 4.

In Table 4, $x_k^{\text{CGKT}}$ and $x_k^{\text{New}}$ represent the iteration points of Algorithm INB with CGKT and New forcing terms, respectively. From Table 4, we see that, in each nonlinear iteration step, the GMRES iterations under CGKT forcing term are more than those under New forcing term. In particular, for each of the first 6 nonlinear iterations, there is only 1 GMRES iteration under New forcing terms while it needs at least 3 GMRES iterations under CGKT forcing terms. However, if comparing $\|F(x_k)\|$ for the first 4 nonlinear steps, we see that the values of $\|F(x_k)\|$ for the two forcing terms are almost the same or differ unclearly. In addition, comparing $\|F(x_k)\|$ for $k = 5$ and 6, we see that the values of New forcing term are smaller than those of CGKT. This shows that CGKT forcing term causes "oversolving" phenomenon. That is, more computational cost is used, but less obtained because the forcing term is too small. In particular, when $k = 5$, we see that 11 GMRES iterations are needed under the CGKT forcing terms, and 1 backtracking is implemented. For the last several nonlinear iterations, the GMRES iterations are all relatively large under both of the forcing terms. Since $r_k > 0.7$ for all $k$ under New forcing term, so, by the new strategy, $\bar{\eta}_k$ is decreased by half in each iteration. The last column of the table clearly shows that the sequence generated under the two forcing terms are very near and converge to the same solution.

## 4. Conclusion

Forcing terms play a very important role in inexact Newton method. They have a strong influence on the efficiency and robustness of the method. To improve the efficiency as well as robustness of inexact Newton method, we propose a new strategy to choose forcing terms. The new forcing terms are determined by $r_k$, the ratio of actual reduction to predicted reduction. They can reflect both the agreement between the local linear model and nonlinear model at the previous step and the reduction of $\|F(x)\|$ in some degree. With the new forcing terms, inexact Newton method is locally $Q$-superlinearly convergent. Numerical results show that the inexact Newton method with this new choice of forcing terms is much more efficient than the method with some old choices, including those given by Eisenstat and Walker [11]. In addition, numerical experiments also show that the local linear model and nonlinear model agree relatively well under the new forcing terms.

## References

[1] H.-B. An, On convergence of the additive schwarz preconditioned inexact newton method, SIAM J. Numer. Anal. 43 (2005) 1850–1871.
[2] S. Bellavia, B. Morini, A globally convergent Newton-GMRES subspace method for systems of nonlinear equations, SIAM J. Sci. Comput. 23 (2001) 940–960.
[3] P.N. Brown, Y. Saad, Hybrid Krylov methods for nonlinear systems of equations, SIAM J. Sci. Statist. Comput. 11 (1990) 450–481.
[4] P.N. Brown, Y. Saad, Convergence theory of nonlinear Newton–Krylov algorithms, SIAM J. Optim. 4 (1994) 297–330.
[5] X.-C. Cai, W.D. Gropp, D.E. Keyes, M.D. Tidriti, Newton–Krylov–Schwarz methods in CFD, in: Proceedings of the International Workshop on Numerical Methods for the Navier–Stokes Equations, Vieweg, Braunschweig, 1995, pp. 17–30.
[6] X.-C. Cai, D.E. Keyes, Nonlinearly preconditioned inexact Newton algorithms, SIAM J. Sci. Comput. 24 (2002) 183–200.
[7] R.S. Dembo, S.C. Eisenstat, T. Steihaug, Inexact Newton methods, SIAM J. Numer. Anal. 19 (1982) 400–408.
[8] R.S. Dembo, T. Steihaug, Truncated Newton algorithms for large-scale optimization, Math. Programming 26 (1983) 190–212.
[9] J.E. Dennis Jr., R.B. Schnabel, Numerical Methods for Unconstrained Optimization and Nonlinear Equations, Prentice-Hall, Englewood Cliffs, NJ, 1983.
[10] S.C. Eisenstat, H.F. Walker, Globally convergent inexact Newton methods, SIAM J. Optim. 4 (1994) 393–422.
[11] S.C. Eisenstat, H.F. Walker, Choosing the forcing term in an inexact Newton method, SIAM J. Sci. Comput. 17 (1996) 16–32.
[12] D.R. Fokkema, G.L.G. Sleijpen, H.A. van der Vorst, Accelerated inexact Newton schemes for large systems of nonlinear equations, SIAM J. Sci. Comput. 19 (1998) 657–674.
[13] I.E. Kaporin, 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.
[14] D.A. Knoll, D.A. Keys, Jacobian-free Newton–Krylov method: a survey of approaches and applications, J. Comput. Phys. 193 (2004) 357–397.
[15] L. Luksan, Inexact trust region method for large sparse systems of nonlinear equations, J. Optim. Theory Appl. 81 (1994) 569–591.
[16] Z. Lužanin, D. Herceg, N. Krejić, Parameter selection for inexact Newton method, Nonlinear Anal. Theory Appl. 30 (1997) 17–24.
[17] J.M. Ortega, W.C. Rheinboldt, Iterative Solution of Nonlinear Equations in Several Variables, SIAM, Philadelphia, PA, 2000.
[18] M. Pernice, H.F. Walker, NITSOL: a Newton iterative solver for nonlinear systems, SIAM J. Sci. Comput. 19 (1998) 302–318.
[19] W.C. Rheinboldt, Methods for Solving Systems of Nonlinear Equations, SIAM, Philadelphia, PA, 1998.
[20] Y. Saad, Iterative Methods for Sparse Linear Systems, SIAM, Philadelphia, PA, 2003.
[21] Y. Saad, M.H. Schultz, GMRES: a generalized minimal residual algorithm for solving nonsymmetric linear systems, SIAM J. Sci. Comput. 7 (1986) 856–869.
[22] J.N. Shadid, R.S. Tuminaro, H.F. Walker, An inexact Newton method for fully coupled solution of the Navier–Stokes equations with heat and mass transport, J. Comput. Phys. 137 (1997) 155–185.
[23] R.S. Tuminaro, H.F. Walker, J.N. Shadid, On backbracking failure in Newton-GMRES methods with a demonstration for the Navier–Stokes equations, J. Comput. Phys. 180 (2002) 549–558.