A modified descent direction for Newton-GMRES method. *

Márcia A. Gomes-Ruggiero Véra L. Rocha Lopes
Julia V. Toledo-Benavides †

Abstract

We consider general Newton-Krylov methods with a line search for solving $F(x) = 0$. In order to curb a possible increase in $\|F\|$, typically occurring during the first few cycles, we propose a simple modification of the Newton direction which does not require a modified Krylov procedure.

Keywords. inexact Newton method, nonlinear systems, Krylov subspace methods, GMRES, backtracking.

AMS subject classifications: 65H10.

1 Introduction

Let $\{x_k\}$ be the iterate sequence of an inexact Newton method [7] for solving

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

where $D \subset \mathbb{R}^n$ is an open and convex set and $F \in \mathcal{C}^1(D, \mathbb{R}^n)$. We have

$$x_{k+1} = x_k + s_k \tag{2}$$

where s_k are approximate solutions for the linear system

$$J(x_k)s = -F(x_k) \tag{3}$$

*This work was supported by FAPESP (Grants 2001-04597-4, 2001-07987-8), CNPq (Grants 140710/99-0, 300603/99-1) and CAPES.

†Department of Applied Mathematics, IMECC-UNICAMP, University of Campinas, CP 6065, 13083-970 Campinas SP, Brazil. e-mails: marcia@ime.unicamp.br, vlopes@ime.unicamp.br and juliatoledob@yahoo.com.br

satisfying the stopping criterion

$$\|J(x_k)s + F(x_k)\| \leq \eta_k \|F(x_k)\|. \tag{4}$$

Here $J(x)$ represents the Jacobian matrix for F at x_k, and the tolerance $\eta_k \in (0, 1]$ is called *the forcing term* [7].

The forcing term has a crucial role in controlling the convergence rate. Several choices for η_k have been proposed (see [12] and [14]). The local convergence analysis for inexact Newton methods shows that if x_0 is sufficiently close to a solution x_* of (1) and the sequence η_k is uniformly bounded away from one, then the sequence $\{x_k\}$ converges to x_* [7].

One of the most popular methods for solving (4) is the Generalized Minimum Residual method (GMRES), [23], which belongs to the family of Newton-Krylov method [6]. A modified version called GMRES(m), or restarted GMRES, is used in large scale problems. Whereas the restart policy is computationally more feasible, convergence cannot be guaranteed in general, and stagnation becomes possible [15], [17], [24], [25], [26] and [28].

The Krylov method requires only the action of the Jacobian J on a vector v. Moreover, for an appropriately chosen scalar ϵ this action can be approximated by finite differences [4]

$$J(x)v \sim \frac{1}{\epsilon} \left[F(x + \epsilon v) - F(x)\right] \tag{5}$$

giving rise to what is known as the matrix-free formalism.

Line search procedures or trust region techniques are used in inexact Newton methods in order to enhance convergence from an arbitrary starting point, see [6], [11], [12], [18], [22]. We follow the line search proposed in [1] and [9] which is a non-monotone strategy similar to the one introduced by Li and Fukushima [13].

In this work we propose a safeguard that modifies the Newton-Krylov line search when a sharp increase in the norm of F is detected. The modification is restricted to the first few iterations, and thus enjoys the same global convergence and robustness properties of the unmodified algorithm. The modified algorithm is also consistent with preconditioning and with matrix-free implementation.

The main advantage of the proposed modification lies in its simplicity and wide application (for example, GMRES can easily be replaced by other linear solvers that give a descent direction). At the same time, it appears that the modified algorithm offers considerable reduction in iteration and time count for some classes of problems.

In Section 2 we briefly review the Newton-GMRES algorithm and the inexact Newton method with backtracking. In Section 3 we present the modified line search and in Section 4 we discuss the implementation of the resulting modified Newton–Krylov method. In Section 4 we also test its performance on two problems from [19] and a set of boundary value problems, showing considerable improvement in some of them, using the performance profile analysis of Dolan and Moré [10]. Concluding remarks are given in subsection 4.3.

2 Preliminaries

2.1 Newton-GMRES

The GMRES method was proposed in [23] for solving linear systems $As = b$, where A is a nonsingular $n \times n$ matrix (not necessarily symmetric) and $b \in I\!R^n$. If s_0 is the initial approximation for the solution and $r_0 = b - As_0$ is the corresponding residual vector, the GMRES Krylov subspace after m iterations will be:

$$\mathcal{K}_m = [r_0\ , Ar_0\ , A^2 r_0\ , \ldots , A^{m-1} r_0]. \tag{6}$$

At the mth GMRES iteration a vector $s_m \in s_0 + \mathcal{K}_m$ is calculated so as to minimize the residual vector, namely, $r_m = \min_{s\ \in\ s_0+\mathcal{K}_m} \|b - As\|$. In what follows we shall exclusively use the 2-norm. When GMRES is used as a linear solver for the Newton method (1), (3), the resulting method is called Newton-GMRES (Algorithm 1 below).

`Algorithm 1.` (GMRES for the kth equation):
Let $x_k,\ \eta_k$ be given.
`Step 1:` Choose s_k^0. Set $m = 0$.
Compute $r_k^0 = -J(x_k)s_k^0 - F(x_k),\ \beta_k = \|r_k^0\|,\ v_1 = r_k^0/\beta_k$.
`Step 2:` While $\|r_k^m\| > \eta_k \|F(x_k)\|$ do
GMRES iteration:
`2.1:` Set $m = m + 1$.
`step 2.2:` Compute $J(x_k)v_m$ and

$$\begin{aligned} &h_{i,m} = (J(x_k)v_m)^T v_i, \quad i = 1, 2, ..., m, \\ &v_{m+1} = J(x_k)v_m - \textstyle\sum_{i=1}^{m} h_{i,m} v_i, \\ &h_{m+1,m} = \|v_{m+1}\|, \\ &v_{m+1} = v_{m+1}/h_{m+1,m}. \end{aligned}$$

Let $\bar{H}_m \in I\!R^{(m+1)\times m}$ be the upper Hessenberg matrix whose nonzero entries are the coefficients $h_{i,j},\ \ i = 1, ..., j+1$, for $j = 1, ..., m$.
`step 2.3:` Find the vector $y_m \in I\!R^m$ that solves the least-squares problem

$$\min_{y \in I\!R^m} \|\beta_k e_1 - \bar{H}_m y\|.$$

`step 2.4:` Set $\|r_k^m\| = \|\beta_k e_1 - \bar{H}_m y_m\|$.
`step 3:` Define $V_m \equiv [v_1,\ v_2, ..., v_m] \in I\!R^{n\times m}$ and form

$$s_k^m = s_k^0 + V_m y_m.$$

`step 4:` Set $s_k = s_k^m$.

We shall henceforth distinguish between *inner iterations* (within the GMRES cycle), denoted by the superscript m, and outer iterations of the Newton algorithm, denoted by the subscript k. At each inner iteration, s_k^m solves the least-squares problem

$$\min_{s \in s_k^0 + K_m} \|J(x_k)s + F(x_k)\|. \tag{7}$$

Step 2.2 is the Arnoldi process [18] for the construction of the orthonormal basis $\{v_i\}$ of K_m. From this process it follows that

$$J(x_k)V_m = V_{m+1}\bar{H}_m, \tag{8}$$

[23], and so the least-squares problem (7) is reduced to

$$\min_{y \in I\!R^m} \|\beta_k e_1 - \bar{H}_m y\|. \tag{9}$$

One keeps iterating until the residual vector $r_k^m = -J(x_k)s_k^m - F(x_k)$ satisfies $\|r_k^m\| \leq \eta_k \|F(x_k))\|$, i.e., until $s_k = s_k^m$ satisfies the stopping criterion (4). Then, the vector s_k is used to form the Newton iterate $x_{k+1} = x_k + s_k$.

It is known that, computationally speaking, GMRES is more expensive than other Krylov subspace methods, such as Bi-CGSTAB, [18], QMR [24] for general square matrices, or LSQR [20], [21] for anti–symmetric matrices. Nevertheless, it is widely used for solving linear systems derived from the discretization of partial differential equations, since in theory the 2–norm of the residual vector is minimized inside the Krylov subspace at each step.

Since the dimension of the Krylov space keeps increasing, the memory cost and complexity of the mth GMRES step increase with m. A modified version called GMRES(m) is used in large scale problems. In this version, the GMRES proceeds in cycles of m iterations, see [18], [23]. The final vector s_m for one cycle is used as the initial vector s_0 for the next cycle; in parallel, $r_m = b - As_m$ is used as initial residue. At each cycle an m–dimensional Krylov subspace is generated from the initial residue, following the usual GMRES procedure.

As mentioned, the restart policy has computational advantages but may lead to stagnation [17], [24], [25], [26] and [28]. A rather expensive remedy would be to monitor the eigenvalues of the Hessenberg matrices generated during the GMRES, [25]. Other schemes, such as the one mentioned in [26], store some vectors created at the jth cycle and use them at the $(j+1)$th cycle.

2.2 Line search

Line search or trust region techniques are often used to obtain global properties for Newton's method, [8]. $d \in I\!R^n$ is called a descent direction of $f(x_k)$ if $f(x_k + \xi d) < f(x_k)$ for $\xi > 0$ small enough. If f is differentiable, this occurs when $\nabla f(x_k)^T d < 0$. It is known that

d is a descent direction for $f(x) = \|F(x)\|^2/2$ if the inequality $\|J(x_k)d+F(x_k)\| < \|F(x_k)\|$ is satisfied (Proposition 3.3 of [6]).

In the case of Newton's method, we conclude from (4) and Algorithm 1 that each outer iterate s_k is a descent direction for f, even if the corresponding residue violates $\|r_k^m\| \leq \eta_k\|F(x_k)\|$. When restarted GMRES is used, s_k is not always a descent direction.

The step length ξ is typically chosen by backtracking, starting with $\xi = 1$ and repeatedly decreasing ξ until f is sufficiently small [8], [11]. Several backtracking methods were formulated for improving convergence of inexact Newton methods from arbitrary starting points, see [5], [6], [11]. In these papers, a monotone decrease of f is used to activate the line search. For example, in [2] and [22] the line search is activated only if the condition

$$\|F(x_k + s_k)\| \leq (1 - t(1 - \eta_k))\|F(x_k)\|, \tag{10}$$

proposed by Eisenstat and Walker, [11] (where $t \in (0,\ 1)$ is fixed) is not satisfied.

Monotone strategies may incur an increase in the function evaluation count due to repeated backtracking, as we indeed found in numerical tests. We preferred to choose a non-monotone line search, due to its increased tolerance during the first iterations. In the technique proposed by Birgin, Krejić and Martinez [1], adopted here, the line search condition (10) is replaced by

$$\|F(x_k + \xi s_k)\| \leq (1 - \xi\sigma)\|F(x_k)\| + \mu_k, \tag{11}$$

where the sequence $\{\mu_k\}$ is such that $\mu_k > 0$ for all $k = 0, 1, 2, \ldots$ and $\sum_{k=0}^{\infty} \mu_k = \mu < \infty$, $\sigma \in (0, 1)$ and $\xi \leq 1$.

Algorithm 2 describes the inexact Newton algorithm with non-monotone line search which we used. Let $x_0 \in I\!R^n$ be an arbitrary initial guess. Given $x_k \in I\!R^n$, and the tolerance $\varepsilon > 0$, x_{k+1} is calculated as follows:

`Algorithm 2.` `NGNLS` (Newton-GMRES method with non-monotone line search):

`While` $\|F(x_k)\| > \varepsilon$, perform steps `1` to `5`:
`Step 1:` Choose η_k.
`Step 2:` Find s_k such that

$$\|F(x_k) + J(x_k)s_k\| \leq \eta_k\|F(x_k)\|; \tag{12}$$

`Step 3:` compute $x_{aux} = x_k + s_k$ and $F(x_{aux})$.
`Step 4:` (backtracking loop) Take $\xi = 1$,
while

$$\|F(x_{aux})\| > [1 - \xi\sigma]\|F(x_k)\| + \mu_k,$$

perform the steps `4.1` and `4.2`:
`step 4.1:` compute $\xi_{new} \in [\varrho_{min}\xi, \varrho_{max}\xi]$;

`step 4.2:` set $\xi = \xi_{new}$ and compute $x_{aux} = x_k + \xi s_k$.
`Step 5:` Take $\xi_k = \xi$, compute $x_{k+1} = x_k + \xi_k s_k$ and update k.

The constants ϱ_{min} and ϱ_{max} are such that $0 < \varrho_{min} < \varrho_{max} < 1$.

3 A new descent direction.

Our aim in this work is to modify Algorithm 2 (NGNLS) by a safeguard which activates a change in the descent direction s_k when a large increase in f is detected. The number of activations is limited, for two reasons: avoiding loss of convergence or robustness; and the empirical fact that a large increase in f typically occurs only in the first few outer iterations. The modified algorithm and direction will be denoted by NDNG (New Direction for Newton-GMRES) and s_b. We make s_b dependent on two parameters: the norm ratio $\|F(x_{aux})\|/\|F(_k)\|$ and the GMRES computational cost.

Ignoring, for the time being, the specific form of the new descent direction, the kth nonlinear step in the modified algorithm is described by Algorithm 3.

`Algorithm 3.` (General formulation):
Let x_k, $\varepsilon > 0$, $L > 0$.
`While` $\|F(x_k)\| > \varepsilon$, perform steps `1` to `4`:
`Step 1:` Choose η_k.
`Step 2:` Find s_k such that (12) is satisfied
`Step 3:` compute $x_{aux} = x_k + s_k$ and $F(x_{aux})$.
`Step 4:` if $\|F(x_{aux})\| > L\|F(x_k)\|$, change s_k by s_b.

Apply the backtracking loop (Step 4, algorithm 2).
`Step 4:` Update x_k.

3.1 Describing the descent direction s_b

As seen in Section 2, line search procedures are based on descent directions for a merit function f whose global minimum is a zero of F; often, $f = \|F\|^2/2$ is used. Let x_k be the current iteration of the Newton-GMRES method. The first restart-GMRES cycle (Step 2 of algorithm 2) starts with $s_k^0 = 0$. In [2] (beginning of section 3.1 there) it is proved that it is possible to extract several descent directions from the GMRES process, for the function f at x_k, without additional cost. This equality is satisfied:

$$\nabla f(x_k)^T v_j = (-\|F(x_k)\|_2) h_{1j}.$$

where h_{1j} is the $(1, j)$ element of the Hessenberg matrix $\bar{H}_m$ and v_j is the jth row vector of

the matrix $V_m : n \times m$, (step 2.2 and 3 of algorithm 1 respectively). Besides $\|F(x_k)\|_2 > 0$ then a vector v_j, $j = 1, \ldots, l$, column vector of matrix V_m, will be a descent direction for f at x_k if $h_{1j} > 0$.

The general idea of our proposal is to consider, besides s_k, an alternative direction, s_d, available through the GMRES process, and then choose $s_b = (1 - \beta)s_k + \beta s_d$, a convex combination, as the new direction. It is required that s_d (hence also s_b) be also a descent direction.

For the first cycle of GMRES, $r_0 = F(x_0)$. A reasonable candidate would be a column v_j of the matrix V_m, provided the $1j$ entry in the matrix $\bar{H}_m$ is strictly positive. We shall choose as s_d the first column of this type. We choice j from $j = m, m-1, ..., 1$.

Meanwhile, we want to choose β as dependent on the inner iterations count as well as the rate of increase of $\|F\|$. To understand better this dependence, look at Fig. 1. Assuming the norm increase $\|F(x_{aux})\| > \|F(x_k)\|$, we get a triangle with height $a_k = \log \|F(x_{aux})\| - \log \|F(x_k)\| > 0$ and base $b_k = \max\{\log(\text{iterin}_k), 1\} > 0$. Here, iterin_k is the cumulative number of inner iterations since the onset of the GMRES cycle of the kth iteration of Algorithm NGNLS. It is reasonable to consider β as a function of the angle ξ_k in this triangle. When $\xi_k \sim \pi/2$, indicating a large increase in $\|F\|$ and fewer iterations, the weight β should favor s_d. The function $\text{sen}(\xi_k)$, where $\text{sen}(\xi_k) = a_k/\sqrt{a_k^2 + b_k^2}$, is consistent with this objective.

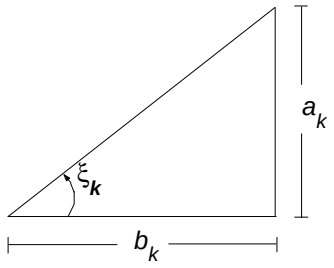

Figure 1: Geometric motivation for the weight β

Actually, we prefer to work with $\beta = \text{sen}^2(\xi_k)$, in order to avoid unnecessary root calculations and based on empirical performance data. Thus, s_b and x_{aux} will be defined as

$$s_b = (1 - \text{sen}^2(\xi_k))s_k + \text{sen}^2(\xi_k)s_d, \qquad x_{aux} = x_k + s_b \tag{13}$$

with $\beta \in (0,\ 1)$. As mentioned, s_b is a descent direction for f at x_k as long as s_k and s_d are.

We denote by Algorithm NDNG the version of Algorithm 3 amended by the modification (13) whenever an increase is detected in the norm of F. For maintain convergence properties, this modification is used a finite times. Then, the new version for the algo-

rithm 3 is:

`Algorithm 4.` `NDNG` (New Descent direcction in Algorithm 3-Newton GMRES):

Let $x_k,\ \varepsilon > 0,\ \ L_1, L_2 > 0,\ \ C = 0.$
`While` $\|F(x_k)\| > \varepsilon$, perform steps `1` to `4`:
`Step 1:` Choose η_k.
`Step 2:` Find s_k such that (12) is satisfeito
`Step 3:` compute $x_{aux} = x_k + s_k$ and $F(x_{aux})$.
`Step 4:` if $\frac{\|F(x_{aux})\|}{\|F(x_k)\|} > L_1$ and $C < L_2$,
`step 4.1:` $s_k = s_b$ from (13).
`step 4.2:` $C = C + 1$.
`step 4.3:` Apply the backtracking loop (Step 4, algorithm 2).
`Step 5:` Update x_k.

Pre-conditioning techniques are commonly used, in order to increase the convergence rate of Krylov methods. We can observe that this process is consistent with pre-conditioning. The analysis is given in [2]. GMRES is consistent with matrix-free implementation, then NDNG method is too.

3.2 Convergence

The convergence of the usual inexact Newton method with monotone line search is guaranteed by [11], and the modification for non-monotone line search is made in [14].

Lemma 1. *Let x_k be a sequence generated by Algorithm 4. If, for some sequence of indices $Q_0 \in \{0,\ 1,\ 2,\ ...\}$, $\lim_{k\in Q_0} F(x_k) = 0$, then*
$\lim_{k\to\infty} F(x_k) = 0$.
In particular, if x_* is a limit point of x_k such that $F(x_*) = 0$, then every limit point of the sequence x_k is a solution of (1).

Proof (See [1]).

Lemma 2. *Let x_k be a sequence generated by Algorithm 4 and assume that all the limit points of the sequence x_k are solution of (1). Assume also that x_* is a limit point of x_k such that $J(x_*)$ is nonsingular and*
$\lim_{k\to\infty} \|(x_{k+1}) - x_k\| = 0$.
Then the whole sequence converges to x_*

Proof (See [1]).

Theorem 1. *Assume that the sequence x_k is generated by Algorithm* 4. *and that there exists $M > 0$ such that, for an infinite sequence of indices $Q_1 \in \{0,\ 1,\ 2,\ ...\}$,*

$$\|J(x_k)s_k + F(x_k)\| \leq \eta_k \|F(x_k)\| \tag{14}$$

and $\|s_k\| < M$. Then any limit point of the subsequence $\{x_k\}_{k \in Q_1}$ is a solution of the system (1). Moreover, if a limit point of $\{x_k\}$ exists, then $F(x_k) \to 0$ and every limit point of $\{x_k\}$ is a solution to (1).

Proof The proof is similar to proof of Theorem 1 in [1], you can see [14].

The version NDNG covered in this paper differs from that of [14] by a limited number of outer iterations, hence enjoys the same convergence properties. We remark that this convergence is subject to the nice behavior of restart-GMRES, which may be compromised in extreme cases of non-linearity of F.

4 Numerical experiments

For the numerical tests we examine the following choices for the forcing term η_k for Step 1 in the algorithm NDNG:

`Constant`: we chose $\eta_k = 0.1$;

`EW1`: $\eta_k = \dfrac{\|F(x_k) - F(x_{k-1}) - J(x_{k-1})s_{k-1}\|}{\|F(x_{k-1})\|}$ (see Eisenstat and Walker [12]);

`EW2`: $\eta_k = \gamma \left(\dfrac{\|F(x_{k+1})\|}{\|F(x_k)\|} \right)^{\alpha}$, $\gamma \in [0,1]$, $\alpha \in (1,2]$. (see Eisenstat and Walker [12]);

`GLT`: $\eta_k = [1/(k+1)]^{\rho} \cos^2(\theta_k) \dfrac{\|F(x_k)\|}{\|F(x_{k-1})\|}$ (see Gomes-Ruggiero et al, [14]), $\rho = 1.1$ and $-\pi/2 \leq \phi_k \leq 0$.

4.1 Implementation features

We give now more details about the implementation of the algorithms. The implementation details can be found in [27], pages 26 and 57. All the tests were performed in a Pentium III - 1.7 GHz computer, using the software MatLab 6.1.

- Line search procedure:
  if the vector $x_{aux} = x_k + \xi s_k$ does not give an acceptable decrease in the value of the function, in the sense of Step 4 of Algorithm 2, then we compute the new step size as $\xi_{new} = 0.5\xi$. For the parameter σ used in Step 4, we took $\sigma = 10^{-4}$.

- The sequence μ_k:
  we define:
  $$ftip(0) = \|F(x_0)\|,$$
  $ftip(k) = \min\{\|F(x_k)\|, ftip(k-1)\}$, if k is a multiple of 3 and
  $ftip(k) = ftip(k-1)$, otherwise.
  Then, we set:
  $$\mu_k = \frac{ftip(k)}{(k+1)^{1.1}}.$$

- The initial value and safeguards for η:
  for all the choices for η_k we set the initial value $\eta_0 = 0.1$. For the choices `EW1` and `EW2` of [12] and for the choice `GLT`, we take $\eta_k = \min\{\eta_k,\ 0.1\}$ if $k \leq 3$, and $\eta_k = \min\{\eta_k,\ 0.01\}$ if $k > 3$. We also take $\eta_k = 0.1$ when $\phi_k > 0$. At the final iterations we have adopted the safeguard introduced in [22] which can be described as: since the linear model is $F(x) \sim F(x_k) + J(x_k)s$, at the final iterations, we can have: $\|F(x_{k+1})\| \sim \|F(x_k) + J(x_k)s_k\| \leq \eta_k\|F(x_k)\|$. In this case it is important to set η_k such that $\eta_k\|F(x_k)\| \sim \varepsilon$ where ε is the precision required for the nonlinear system. A safeguard which represents these ideas is: if $\eta_k \leq 2\varepsilon$ then we set $\eta_k = (0.8\varepsilon)\|F(x_k)\|$.

- Parameters choice for η_k:
  for the choice `EW2` it was taken $\gamma = 1$ and $\alpha = 0.5(1 + \sqrt{5})$ and for the choice `GLT` it was taken $\rho = 1.1$.

- Stopping criterion:
  the process is finished successfully if $\|F(x_k)\| \leq 10^{-6}$ and $k < 100$.

- Restarts and the maximum number of iterations in GMRES(m):
  we fix the restarts at each 30 iterations, $m = 30$, allowing initially a maximum of 100 cycles (3000 iterations).

4.2 Boundary value problems

The general formulation of the boundary value problems solved in this work is finding $u : \Omega = [\ 0,\ 1] \times [\ 0,\ 1] \to I\!R$, such that, for $\lambda \in I\!R$,

$$-\Delta u + h(\lambda,\ u) = f(s,t),\ \text{in}\ \Omega,\ \ u(s,t) = 0\ \ \text{on}\ \ \partial\Omega. \tag{15}$$

The real valued function $h(\lambda, u)$, the different values for the parameter λ and the function f define the different problems tested. All the problems were discretized using central differences on a grid with 63 inner points in each axis. The discretized system obtained has 3969 equations and variables. We now make a brief description of the particular problems that were solved:

- `Problem 1` - A convection–diffusion problem: in this problem, the function h is given by $h(\lambda, u) = \lambda u(u_s + u_t)$, where u_s and u_t denote the partial derivatives of the function u with respect to s and t, and again the function $f(s,t)$ is defined so that $u^*(s,t) = 10st(1-s)(1-t)e^{s^{4.5}}$ is the exact solution for the problem. This is a problem considered difficult to solve [16], in particular for values of λ greater than 50.

- `Problem 2` - The problem appears in the book of Briggs, Henson and McCormick [3], page 105. In this case, $h(\lambda,\ u)$ is given by $h(\lambda,\ u) = \lambda u e^u$ and the function $f(s,t) = ((9\pi^2 + \gamma e^{(x^2-x^3)\sin(3\pi y)})(x^2 - x^3) + 6x - 2)\sin(3\pi y)$.

Taking in account that the increase in $\|F\|$ is usually limited to the first outer iterations, we chose to limit the number of descent direction modifications to 5 among the first 10 iterations. This modification is triggered by the condition $\|F(x_{aux})\|_2/\|F(x_k)\|_2 > 10$. For computational reasons we modified the calculation of $\mathrm{sen}(\xi_k)$ by a weighting at a_k as follows: if $a_k/b_k \geq 2$ then we set $a_k = ca_k$, where we chose $c = 0.2$, for empirical reasons.

We compared NDNG against the Bellavia and Morini algorithm (algorithm 3 in [2]), henceforth algorithm BM. To this end we chose 2 problems taken from the Lukšan collection [19], for which both strategies converged for most initial data: Problem 4.2 (extended Powell badly scaled function), henceforth Powell with $x_s = (0, 1, \ldots, 0, 1)$; and Problem 4.7 (Tridiagonal system, henceforth Tridiagonal) with $x_s = (12, 12, \ldots, 12)$ where x_s represents the initial value indicated by [19]. For each problem, several initial data were used, including integer multiples of x_s and $ones = (1, \cdots, 1)$ as well as zero initial data. η_k was chosen according to EW2.

The BM data are taken from [2] which uses these parameters. A comparison of the results is shown in Table 1 in terms of the numbers of outer iterations (iterex) and function evaluation (feval). Each pair of numbers represents the results obtained by the two algorithms, in this order: (BM, NDNG).

In this table, `*` indicates the triggering of safeguard 4 in Algorithm 4, and `-` indicates lack of convergence. We observe that in all cases NDNG was better than BM, even if the safeguard 4 was not activated, except where both algorithms failed to converge. In both problems, the reduction in feval is especially notable. It seems that this is primarily due to the introduction of a non-monotone line search.

In Table 2 we compare NDNG against the usual Newton-GMRES method NGNLS in solving Problems 1 (convection-diffusion) with $x_0 = (0, 0, \ldots, 0)$ and 2 (Briggs) with $x_0 = (-2, -2, \ldots, -2)$, both with $\lambda = 100$. Each pair of numbers represents the results obtained by the two algorithms, in this order: (NGNLS, NDNG). We observe a considerable reduction across the board in Problem 1, in which Safeguard 4 was invariably triggered. Only marginal improvement was detected in Problem 2. A comparison of the performance profiles of these algorithms is presented in Figure 2, where both choices `EW2` and `GLT` for η_k were examined.

Table 1: Comparison of BM and NDNG

	Powell		Tridiagonal	
x_0	iterex	feval	iterex	feval
x_s	(135, 104)	(1034, 316)	(58, 39)*	(493, 98)
$2x_s$	(133, 102)	(1029, 306)	(215, 87)	(3061, 391)
$5x_s$	(125, 93)*	(1009, 381)	–	–
$-x_s$	–	–	(15, 17)*	(74, 22)
$-2x_s$	–	–	(17, 17)	(70, 18)
$-5x_s$	–	–	(19, 19)	(85, 22)
$ones$	(133, 102)	(1040, 312)	(0,0)	(0,0)
$2ones$	(134, 99)	(1031, 296)	(9, 9)	(45, 10)
$5ones$	(120, 93)*	(918, 379)	(25, 25)*	(152, 46)
$null$	–	–	(10, 8)*	(58,10)

Here, `EW2` and `GLT` refer to these choices within the NGNLS algorithm, while an extra "d" refers to the modified NDNG version. The following nine problems were considered: 6 variants of Problem 1 with $\lambda = 50,\ 75,\ 100,\ 110,\ 125,\ 150$ and $x_0 = (0, 0, \ldots, 0)$; Problem 2 with $\lambda = 100,\ 1000$ and $x_0 = (-2, -2, \ldots, -2)$; and Problem 2 with $\lambda = 1000$ and $x_0 = (-1, -1, \ldots, -1)$.

Table 2: Comparing NGNLS and NDNG

η_k	problem	iterex	iterin	feval	CPU time
Cte.	conv-dif	(21, 15)	(5567, 2704)	(68, 40)	(256, 156)
	Briggs	(10, 9)	(451, 434)	(13, 13)	(27, 26)
EW1	conv-dif	(20, 15)	(7716, 2641)	(75, 44)	(306, 149)
	Briggs	(9, 8)	(419, 392)	(12, 12)	(28, 23)
EW2	conv-dif	(21, 14)	(7730, 2631)	(75, 44)	(459, 202)
	Briggs	(9, 8)	(375, 381)	(12, 12)	(25, 27)
GLT	conv-dif	(18, 14)	(6164, 2578)	(66, 44)	(239, 130)
	Briggs	(9, 7)	(383, 349)	(12, 11)	(25, 26)

We observe that NDNG, especially with choice GLT, was optimal with respect to all four measures, solving about 50% of the problems with the least value for `iterin, feval, CPU time` and 80% for `iterex`. It was also the most robust with respect to all four measures.

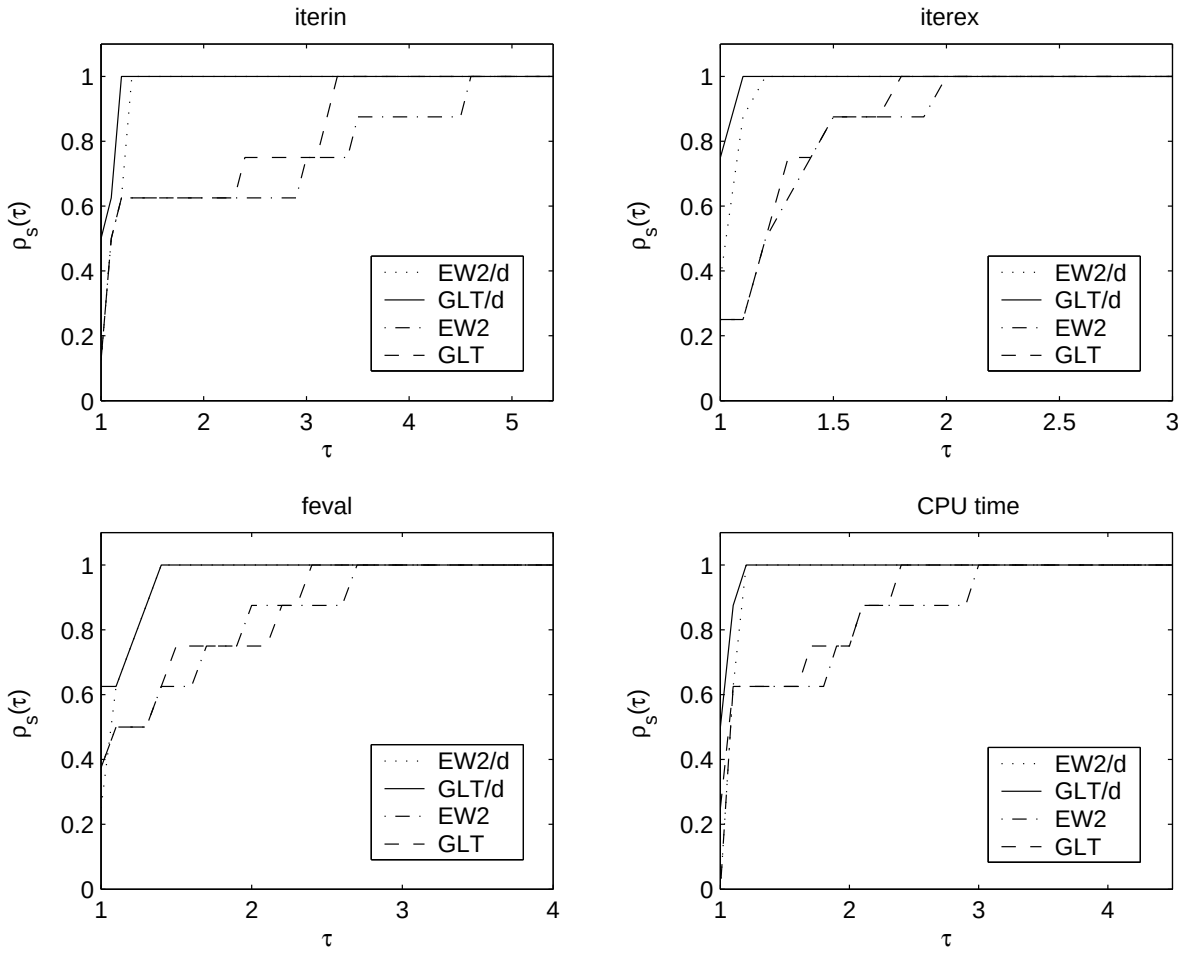

Figure 2: Performance profile of NGNLS and NDNG

4.3 Conclusions

A strategy for improving the Newton-GMRES performance (with non-monotone line search) was introduced. The strategy consists of a modification of the descent direction generated by GMRES when a sharp increase in the norm of F is detected. Only a small number of modifications is allowed, so as not to interfere with the convergence pattern. Numerical results show an improvement relative to the usual Newton-GMRES algorithm, in terms of outer and inner iterations, function evaluations and CPU time. The obtained process is consistent with pre-conditioning and with matrix-free implementation.

References

[1] E. G. Birgin, N. Krejić, and J. M. Martínez, Globally convergent inexact quasi-Newton methods for solving nonlinear systems, *Numerical algorithms* **32** (2003) 249–260.

[2] S. Bellavia and B. Morini, A globally convergent Newton-GMRES subspace method for systems of nonlinear equations, *SIAM J. Sci. Comput.* **23** 3 (2001) 940–960.

[3] W. L. Briggs, V. E. Henson and S. F. McCormick, *A Multigrid Tutorial*, Second Edition, SIAM, Philadelphia, 1987.

[4] P. N. Brown, A local convergence theory for combined inexact-Newton/finite-difference projection methods, *SIAM J. Numer. Anal.* **24** (1987) 407–434.

[5] P. N. Brown, and Y. Saad, Hybrid Krylov methods for nonlinear systems of equations, *SIAM J. Sci. Stat. Comput.* **11** 3 (1990) 450–481.

[6] P. N. Brown and Y. Saad, Convergence theory of nonlinear Newton-Krylov algorithms, *SIAM J. Optim.* **4** (1994) 297–330.

[7] R. S. Dembo, S. C. Eisenstat and T. Steihaug, Inexact Newton methods, *SIAM J. Numer. Anal.* **19** 2 (1982) 401–408.

[8] J. E. Jr. Dennis, and R. B. Schnabel, *Numerical Methods for Unconstrained Optimization and Nonlinear Equations*, SIAM Classics in Applied Mathematics, 1996.

[9] M. A. Diniz-Ehrhardt, M. A. Gomes-Ruggiero, V. L. R. Lopes and J. M. Martínez, Discrete Newton's method with local variations for solving large-scale nonlinear systems, *to appear in Optimization.*

[10] E. D. Dolan and J. J. Moré, Benchmarking optimization software with performance profiles, *Math. Program. Ser.* **A91** (2002) 191–213.

[11] S. C. Eisenstat and H. F. Walker, Globally convergent inexact Newton methods, *SIAM J. Optim.* **4** (1994) 393–422.

[12] S. C. Eisenstat and H. F. Walker, Choosing the forcing terms in inexact Newton method, *SIAM J. Sci. Comput.* **17** 1 (1996) 16–32.

[13] D-H. Li, and M. Fukushima, Derivative-free line search and global convergence of Broyden-like method for nonlinear equations, *Optimization Methods and Software* **13** (2000) 181-201.

[14] M. A. Gomes-Ruggiero, V. L. Rocha Lopes and J. V. Toledo-Benavides, A new choice for the forcing term and a global convergent inexact Newton method, Technical Report - Unicamp **52/03** (2003) - to appear in *Annals of Operations Research*.

[15] M. A. Gomes-Ruggiero, V. L. Rocha Lopes and J. V. Toledo-Benavides,Some applications for Newton-Krylov methods with a safeguard for GMRES(m). Relatório RP2/06, IMECC, UNICAMP, 02/2006.

[16] M. A. Gomes–Ruggiero, D. N. Kozakevich and J. M. Martínez, A numerical study on large–scale nonlinear solvers, *Computers and Mathematics with Applications* **32** 3 (1996) 1–13.

[17] W. Joubert, On the convergence behavior of the restarted GMRES algorithm for solving nonsymmetric linear systems, *Numerical Linear Algebra with Applications* **1** 5 (1994) 427-447.

[18] C. T. Kelley, *Iterative methods for linear and nonlinear equations*, SIAM, 1995.

[19] L. Lukšan, Inexact trust region method for large sparse systems of nonlinear equations, *Journal of Optimization Theory and Applications* **81** (1994) 569–590.

[20] C. C. Paige and M. A. Saunders, Algorithm 583 LSQR: Sparse linear equations and least squares problems, *ACM Trans. Math. Software* **8** (1982) 195–209.

[21] C. C. Paige and M. A. Saunders, LSQR: An algorithm for sparse linear equations and sparse least squares, *ACM Trans. Math. Software* **8** (1982) 43–71.

[22] M. Pernice, and H. F. Walker, NITSOL: a Newton iterative solver for nonlinear systems, *SIAM J. Sci. Comput.* **19** 1 (1998) 302–318.

[23] Y. Saad and M. H. Schultz, GMRES: A generalized minimal residual algorithm for solving nonsymmetric linear system, *SIAM J. SCI. Comput.* **7** 3 (1986), 856–869.

[24] Y. Saad, *Iterative Methods for Sparse Linear Sistems*, The PWS Publishing Company, 1996.

[25] V. Simoncini, On the Convergence of restarted Krylov subspace methods, *SIAM J. Matrix Anal. Appl.* **22** 4 (2000) 430–452.

[26] E. de Sturler, Truncation strategies for optimal Krylov subspace methods, *SIAM J. Numer. Anal.* **36** (1999) 864–889.

[27] J. V. Toledo-Benavides, A globally convergent Newton-GMRES method with a new choice for the forcing term and some strategies to improve GMRES(m), *PhD Thesis, T/UNICAMP T575m,* (2005).

[28] I. Zavorin, D. P. O'Leary and H. Elman, Complete stagnation of GMRES, *Linear Algebra and its Applications*, **367** (2003) 165–183.