An International Journal of Optimization and Control: Theories & Applications
ISSN:2146-0957 eISSN:2146-5703
Vol.9, No.2, pp.176-185 (2019)
http://doi.org/10.11121/ijocta.01.2019.00743

RESEARCH ARTICLE

A New Broyden rank reduction method to solve large systems of nonlinear equations

Mostafa Ouarit [a], Ali Souissi[b] and Mohammed Ziani*[b]

[a] *MACS Laboratory, Department of Mathematics and Computer Sciences, Faculty of Sciences Ain Chock, Hassan II University Of Casablanca, Morocco*
[b] *Numerical Analysis Group, LMSA, CeReMAR, Department of Mathematics, Faculty of Sciences, Mohammed V University in Rabat, Morocco*
mostafa.ouarit@univh2c.ma, souissi@fsr.ac.ma, ziani@fsr.ac.ma

ARTICLE INFO

Article History:
Received 04 November 2018
Accepted 03 February 2019
Available 09 April 2019

Keywords:
Large nonlinear systems of equations
Limited memory Broyden method
Singular values thresholding
Rank reduction
Superlinear convergence

AMS Classification 2010:
65H10; 65Y20; 15A15

ABSTRACT

We propose a modification of limited memory Broyden methods, called dynamical Broyden rank reduction method, to solve high dimensional systems of nonlinear equations. Based on a thresholding process of singular values, the proposed method determines *a priori* the rank of the reduced update matrix. It significantly reduces the number of singular values decomposition calls of the update matrix during the iterations. Local superlinear convergence of the method is proved and some numerical examples are displayed.

1. Introduction

Let us consider the problem of finding a solution of the system of nonlinear equations

$$F(x) = 0, \qquad F : \mathbb{R}^n \to \mathbb{R}^n. \tag{1}$$

The mapping F is assumed to fulfill the following classical assumptions (CA):

- it is continuously differentiable in an open convex set $\mathcal{D} \subset \mathbb{R}^n$,
- there is an x_* in $\mathcal{D}$ such that $F(x_*) = 0$,
- the Jacobian F' is Lipschitz continuous at x_* and $F'(x_*)$ is nonsingular.

Newton's method (see [1–3])

$$x_{k+1} = x_k - (F'(x_k))^{-1}F(x_k), \quad k = 0, 1, \ldots, \tag{2}$$

is well suited to solve the system (1) due to its local quadratic convergence. However, this method is known to be numerically expensive. It requires the evaluation of a jacobian matrix and the solution of a linear system per iteration. An alternative to Newton's method is the Broyden's quasi-Newton method. This method uses approximations to the Jacobian matrix at each iteration by performing rank-one updates, see [4]. It requires only one F-evaluation per iteration and achieves, under the classical hypotheses (CA), local superlinear convergence as shown in [5]. Given an initial guess x_0 and an initial approximation B_0 of the jacobian matrix, Broyden iteration is given by

$$x_{k+1} = x_k - B_k^{-1}F(x_k), \quad k = 0, 1, \ldots, \tag{3}$$

where B_k is updated at each iteration as

$$B_{k+1} = B_k + (y_k - B_k s_k)\frac{s_k^T}{s_k^T s_k}, \tag{4}$$

*Corresponding Author

with $y_k = F(x_{k+1}) - F(x_k)$ and $s_k = x_{k+1} - x_k$. To avoid the drawback of storing and manipulating the $(n\times n)$-matrices of Broyden, limited memory methods put restrictions on the size of systems to solve, see references [6], [7] and [8]. Equation (4) implies that if B_0 is updated p times, the resulting matrix can be written as

$$B_p = B_0 + c_1 d_1^T + \ldots + c_p d_p^T = B_0 + CD^T, \quad (5)$$

where

$$c_{j+1} = \frac{y_j - B_j s_j}{\| s_j \|}$$

and

$$d_{j+1} = \frac{s_j}{\| s_j \|},\ j = 0, ..., p-1,$$

with $C = [c_1, ..., c_p]$ and $D = [d_1, ..., d_p]$. The update matrix $Q = CD^T$ is sum of p rank-one matrices, its rank does not exceed p. The singular values decomposition of Q is then given by

$$Q = \sigma_1 u_1 v_1^T + \ldots + \sigma_p u_p v_p^T, \quad (6)$$

where $\sigma_1 \geq \ldots \geq \sigma_p \geq 0$ are the singular values. The sets $\{u_1, \ldots, u_p\}$ and $\{v_1, \ldots, v_p\}$ are, respectively, the left and right corresponding singular vectors. By choosing $B_0 = I$, the matrix B_p can be stored using $2p$ vectors of length n.

Suppose the maximal rank of Q is fixed at p with $p << n$. Broyden rank reduction (BRR) method, see [7], is a variant of limited memory Broyden methods that approximates Q by a matrix $\tilde{Q}$ with low rank $q \leq p-1$ by truncating the singular value decomposition (6).

The current number of stored updates is denoted by m $(0 \leq m \leq p)$. The maximal number of updates to the initial Broyden matrix is thus given by p. If p updates are stored $(m = p)$, the singular value decomposition of Q is computed and the last $(p-q)$ singular values are removed just before the next update is computed. In this case, the next Broyden update will proceed using the modified matrix

$$\tilde{B}_k = B_k - R,$$

where $R = \sum_{l=q+1}^{p} \sigma_l u_l v_l^T$ is the so-called round-off matrix. No reduction is performed as long as $m < p$. Thus, the liberated memory is used to store $(p-q)$ new Broyden's updates according to the scheme (7)

$$B_{k+1} = \begin{cases} B_k + (y_k - B_k s_k)\dfrac{s_k^T}{s_k^T s_k} - R\left(I - \dfrac{s_k s_k^T}{s_k^T s_k}\right) & \text{if } m = q+1, \\ B_k + (y_k - B_k s_k)\dfrac{s_k^T}{s_k^T s_k}, & \text{else.} \end{cases} \quad (7)$$

The BRR method, as presented in [7], does not give any idea how to fix *a priori* the rank of the matrix $\tilde{Q}$, only the smallest singular value is removed. But, in many cases there are more than one singular value that are close to zero and so they can be removed. In this case, memory will be free to store more than one Broyden's update. We propose here a new approach using a thresholding process of singular values of the update matrix by fixing a relative accuracy for the approximation of the matrix Q. In section 2 we present the new method and prove its local superlinear convergence. Section 3 is devoted to numerical results showing the efficiency of the method.

2. The proposed method

In many nonlinear problems, the singular values of the update matrix decay rapidly to zero, and more than one singular value can be removed. In figure 3 (see problem 3 in section 3) we present the singular values distribution of Q in case of the Spedicato function for $p = 6$ and $p = 10$. For example, when $p = 6$, we can see that the two last smallest singular values are zero while the third and the fourth ones are close to zero. In this example, the four last singular values can advantageously be removed as memory will be available to store four new Broyden updates and no singular values decomposition will be needed during the following four iterations.

So, the question is how, in general, to choose the rank q of $\tilde{Q}$ and thence the number of singular values to remove. As an answer to this question, we propose to use a tresholding process by exploiting the information about the approximation error $\| R \|_2$. Given a relative accuracy $\varepsilon > 0$ of the approximation $\tilde{Q}$, i.e.,

$$||Q - \tilde{Q}||_2 < \varepsilon ||Q||_2,$$

the required rank $q(\varepsilon)$ is given, if it exists, by

$$q(\varepsilon) = \min\{k \in \{1, \ldots, p-1\} \text{ s.t. } \sigma_{k+1} < \varepsilon\sigma_1\}. \quad (8)$$

The value of $q(\varepsilon)$ is calculated each time a reduction of the update matrix is needed, and all singular values satisfying the condition $\sigma_{l+1} < \varepsilon\sigma_1$, $l = 1, \ldots, p-1$ are removed. If there is no k satisfying (8), only the smallest singular value is removed (we turn back to the BRR method), and in this case $q = p-1$. So, the rank q will be chosen as

$$q = \begin{cases} q(\varepsilon) & \text{if } q(\varepsilon) \text{ exists,} \\ p-1 & \text{otherwise.} \end{cases} \tag{9}$$

This dynamical choice of q leads to a new method, displayed in Algorithm 1, which will be called dynamical Broyden rank reduction (DBRR) method.

In DBRR algorithm, the Sherman Morrison Woodbury formula, see [1], is used in order to compute the inverse of the Broyden matrix. If we set $B_0 = I$ we get

$$(I + CD^T)^{-1} = I - C(I + D^T C)^{-1} D^T.$$

In this case, we have only to solve linear systems of equations with $p \times p$ matrices. Note also that the singular value decomposition of the update matrix is carried out using an economical process, see [7]. Note that the error that is introduced by removing the last $(p-q)$ singular values of Q equals

$$||R|| = \begin{cases} \sigma_{q+1} & \text{for} \quad m = q+2, \\ 0 & \text{else.} \end{cases}$$

Let us now prove the superlinear convergence of the proposed algorithm.

Theorem 1. *Let q be defined as in (9) with*

$$||R|| \leq \alpha \parallel s_k \parallel, \; s_k \neq 0, \; k \in \mathbb{N}, \tag{10}$$

where the constant α does not depend on k. Let, in addition, the classical hypotheses (CA) hold. Then the DBRR method has local superlinear convergence.

Proof. Define $E_k = B_k - F'(x_*)$ and $e_k = x_k - x_*$ for $k = 0, 1, \ldots$. To prove linear convergence of the proposed method, we need the following lemma whose proof can be found in [1], p. 77.

Lemma 1. *Let $F : \mathbb{R}^n \to \mathbb{R}^n$ be continuously differentiable in the open convex set $\mathcal{D} \subset \mathbb{R}^n$ and F' is γ-Lipschitz in $x \in \mathcal{D}$. Then, for any $u, v \in \mathcal{D}$,*

$$\begin{aligned} &\big\| F(v) - F(u) - F'(x)(v-u) \big\| \\ &\leq \frac{\gamma}{2} \left(\|v - x\| + \|u - x\| \right) \|v - u\|. \end{aligned}$$

Equation (7) can be written as

$$\begin{aligned} E_{k+1} \;&=\; E_k \left(I - \frac{s_k s_k^T}{s_k^T s_k} \right) + \big(y_k - F'(x_*) s_k \big) \frac{s_k^T}{s_k^T s_k} \\ &\quad - R \left(I - \frac{s_k s_k^T}{s_k^T s_k} \right). \end{aligned} \tag{11}$$

Using Lemma 1 we have

$$\big\| y_k - F'(x_*) s_k \big\| \leq \frac{\gamma}{2} \left(||e_{k+1}|| + ||e_k|| \right) ||s_k||, \tag{12}$$

where γ is the constant Lipschitz for F'. Thus we obtain, from equation (11), the so-called bounded deterioration property of the DBRR method

$$||E_{k+1}|| \leq ||E_k|| + \left(\alpha + \frac{\gamma}{2} \right) \left(||e_{k+1}|| + ||e_k|| \right), \tag{13}$$

since $||R|| = \sigma_{q+1} \leq \alpha ||s_k|| \leq \alpha \left(||e_{k+1}|| + ||e_k|| \right)$. We have used the fact that $\left(I - \frac{s_k s_k^T}{s_k^T s_k} \right)$ is an orthogonal projection and then its norm is equal to one. Inequality (13) implies local convergence of the DBRR method. In fact, as shown in [5], any quasi-Newton method that obeys the bounded deterioration property has local linear convergence. As consequences of the linear convergence, we have

$$||e_{k+1}|| \leq \frac{1}{2} ||e_k||, \quad k = 0, 1, \ldots, \tag{14}$$

and

$$\sum_{k=0}^{\infty} ||e_k|| \leq 2 ||e_0||. \tag{15}$$

Since DBRR method satisfies the secant equation $(B_{k+1} s_k = y_k)$, according to Theorem 8.2.4 in [1], a sufficient condition for $\{x_k\}_k$ to converge superlinearly to the root x_* is the so-called Dennis-Moré condition

$$\lim_{k \to +\infty} \frac{||E_k s_k||}{||s_k||} = 0. \tag{16}$$

Equation (11) also implies

$$\begin{aligned} ||E_{k+1}||_F \;&\leq\; \left\| E_k \left(I - \frac{s_k s_k^T}{s_k^T s_k} \right) \right\|_F \\ &\quad + \left\| \big(y_k - F'(x_*) s_k \big) \frac{s_k^T}{s_k^T s_k} \right\|_F + ||R||_F, \end{aligned}$$

Algorithm 1. *Let $x_0 \in \mathbb{R}^n$ and $B_0 \in \mathbb{R}^{n\times n}$ be given. Set the parameter p and the accuracy $\varepsilon > 0$. Let $m = 0$.*

: *For $k = 0, 1, 2, \ldots$ until convergence,*

1. : *Solve $B_k s_k = -F(x_k)$ for s_k*

2. : $x_{k+1} = x_k + s_k$

3. : $y_k = F(x_{k+1}) - F(x_k)$

4. : *If $m = p$ then*

4.1. : *Compute the singular values decomposition of Q as in (6)*

4.2. : *Compute q as in (9)*

4.3. : *Reduce Broyden matrix:* $B_k = B_k - \sum_{i=q+1}^{p} \sigma_i u_i v_i^T$

4.4. : *Set $m = q$*

5. : *Update matrix B_k as in (7)*

6. : *Set $m = m + 1$*

where $||.||_F$ denotes the Frobenius norm. From equation (12), we obtain

$$\begin{aligned} & \left\| \left(y_k - F'(x_*)s_k\right) \frac{s_k^T}{s_k^T s_k} \right\|_F \\ = & \left\| \left(y_k - F'(x_*)s_k\right) \frac{s_k^T}{s_k^T s_k} \right\| \\ \leq & \frac{\gamma}{2}\left(||e_{k+1}|| + ||e_k||\right). \end{aligned}$$

The following lemma, where the proof can be found in [1], p. 183, will be useful in the sequel.

Lemma 2. *Let $s \in \mathbb{R}^n$ be nonzero and let $E \in \mathbb{R}^{n\times n}$. Then*

$$\left\| E\left(I - \frac{ss^T}{s^T s}\right) \right\|_F \leq \|E\|_F - \frac{1}{2\|E\|_F}\left(\frac{||Es||}{||s||}\right)^2.$$

Using inequality (14) and Lemma 2 we derive

$$\begin{aligned} ||E_{k+1}||_F & \leq ||E_k||_F - \tfrac{||E_k s_k||^2}{2||E_k||_F ||s_k||^2} \\ & + \tfrac{3}{4}\gamma||e_k|| + ||R||_F \\ & \leq ||E_k||_F - \tfrac{||E_k s_k||^2}{2||E_k||_F ||s_k||^2} \\ & + \tfrac{3}{4}\left(\gamma + 2\alpha\sqrt{n}\right)||e_k||. \end{aligned}$$

This inequality is equivalent to

$$\frac{||E_k s_k||^2}{||s_k||^2} \leq 2||E_k||_F \left(||E_k||_F - ||E_{k+1}||_F + \frac{3}{4}(\gamma + 2\alpha\sqrt{n})||e_k||\right). \quad (17)$$

Using inequality (15) we show that

$$\sum_{k=0}^{+\infty} \frac{||E_k s_k||^2}{||s_k||^2} \leq 2c\left(||E_0||_F + \frac{3}{2}(\gamma + 2\alpha\sqrt{n})||e_0||\right),$$

where $c > 0$ is an upper bound of the sequence $\{E_k\}_k$. Hence, condition (16) is satisfied and the superlinear convergence is proved. □

3. Numerical results

We present now numerical tests by applying the proposed method to some classical test functions from the literature and we present a comparison of this method with the classical BRR method ($q = p - 1$). The numerical experiments were carried out using the scientific computing software MATLAB. We use the following stopping criterion for our computer programs

$$||F(x_k)|| < \varepsilon_a + \varepsilon_r ||F(x_0)||,$$

where $\varepsilon_a = 10^{-15}$ and $\varepsilon_r = 10^{-15}$ (respectively, absolute and relative tolerances). For both BRR and DBRR methods most of the computational time is spent in evaluations of the function F and computation of the singular values decomposition of the update matrix.

Problem 1

Let us consider the trigonometric system

$$\begin{cases} F_1(x) = \cos(x_1) - 9 + 3x_1 + 8\exp(x_2), \\ F_i(x) = \cos(x_i) - 9 + 3x_i + 8\exp(x_{i-1}), \quad i = 2, \ldots, n-1, \\ F_n(x) = \cos(x_i) - 1. \end{cases}$$

The size of this problem is $n = 1000000$, and the initial guess is given by $x_0 = (1.2, \ldots, 1.2)^T$. We plot in figure 1 the distribution of singular values of the update matrix for $p = 5, 8, 10$ and

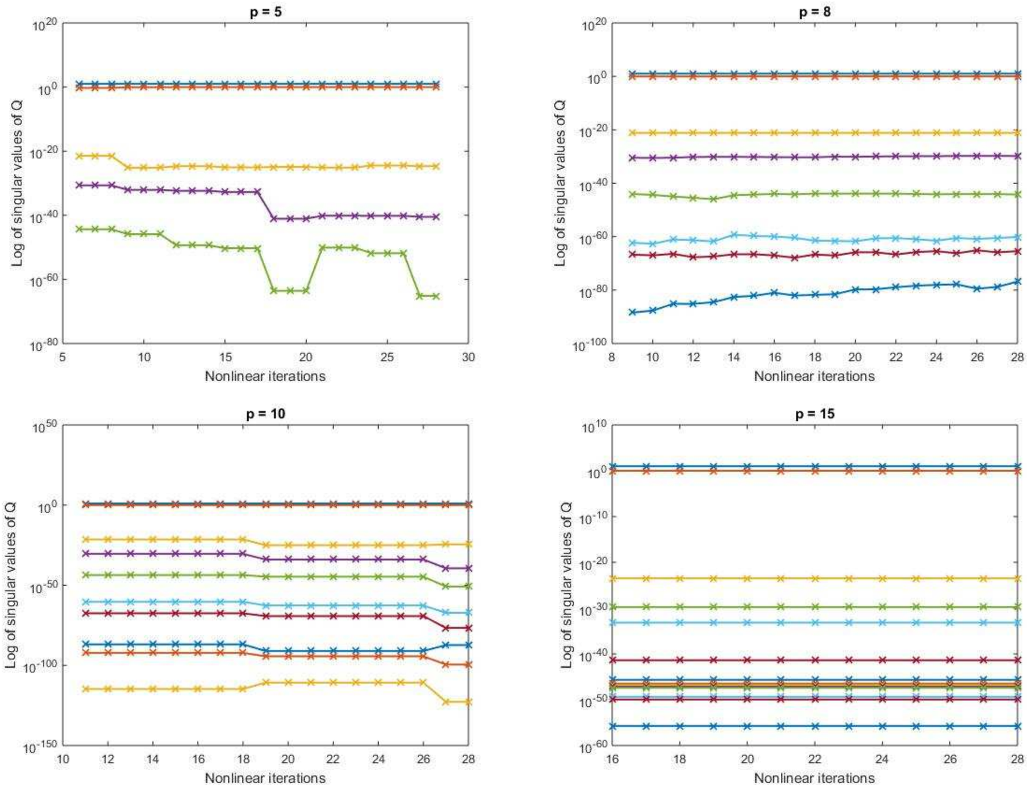

Figure 1. Singular values of the update matrix Q for $p = 5, 8, 10$ and $p = 15$ for Problem 1.

$p = 15$. For these values of p, the update matrix has rank two in all nonlinear iterations. Hence, the DBRR method requires less singular values decomposition calls. The size of this problem is $n = 1000000$, and the initial guess is given by $x_0 = (1.2, \ldots, 1.2)^T$. We plot in figure 1 the distribution of singular values of the update matrix for $p = 5, 8, 10$ and $p = 15$. For these values of p, the update matrix has rank two in all nonlinear iterations. Hence, the DBRR method requires less singular values decomposition calls.

For this example both BRR and DBRR do not converge for $p \leq 4$. Performances of DBBR and BRR methods, for different values of p, are presented in tables 1 and 2, respectively. For all p values, the choice of ε does not significantly affect the convergence rate and the computational time.

Problem 2

We consider now the so-called extended system of Byeong

$$F_i(x) = \cos(x_i^2 - 1) - 1, \quad i = 1, \ldots, n.$$

The size of this problem is $n = 1000000$, and the initial guess is given by $x_0 = (0.0087, \ldots, 0.0087)^T$. For this example, the update matrix has rank one during all nonlinear iterations, see figure 2.

So, whenever a reduction of the update matrix is needed, $(p - 1)$ singular values are removed freeing the memory to store new updates. For a given value of p, the choice of the parameter ε does not affect the performance of DBRR method as shown in table 3. A comparison of tables 3 and 4 shows the efficiency of the singular values thresholding process.

Problem 3

In this example, we compute the root of the so-called Spedicato function

$$F_i(x) = \begin{cases} 1 - x_i & \text{if} \quad i \quad \text{odd,} \\ 10(x_i - x_{i-1}^2) & \text{if} \quad i \quad \text{even,} \end{cases}$$

for $i = 1, \ldots, n$. The size of this problem is $n = 1000000$, and the initial guess is given by $x_0 = (-1.2, \ldots, -1.2)^T$. Both BRR and DBRR methods do not converge for $p \leq 4$. The rank of Q increases with the nonlinear iterations as shown in figure 3.

Performances of DBBR and BRR methods are presented in tables 5 and 6, respectively.

Problem 4

We consider the nonlinear convection-diffusion partial differential equation

Table 1. Performance of the DBRR method for Problem 1.

	$p=5$				$p=8$			
$\varepsilon =$	10^{-2}	10^{-4}	10^{-6}	10^{-10}	10^{-1}	10^{-3}	10^{-5}	10^{-10}
Iters	28	28	28	28	28	28	28	28
CPU time	10.865	11.104	12.011	11.021	10.380	12.417	12.879	11.505
SVD calls	8	8	8	8	3	4	4	4
SVD time	2.961	3.106	3.338	3.024	1.435	3.308	3.386	2.999
% SVD time	27.2	28.0	27.7	27.5	13.8	26.6	26.3	26.1
	$p=10$				$p=15$			
$\varepsilon =$	10^{-1}	10^{-3}	10^{-5}	10^{-10}	10^{-1}	10^{-3}	10^{-5}	10^{-10}
Iters	28	28	28	28	28	28	28	28
CPU time	10.989	13.712	13.235	12.897	14.436	13.227	13.224	13.269
SVD calls	2	3	3	3	1	1	1	1
SVD time	1.296	3.713	3.244	3.233	1.436	1.220	1.200	1.220
% SVD time	11.8	27.1	26.0	25.0	9.9	9.8	9.8	9.9

Table 2. Performance of the BRR method for Problem 1.

p	5	8	10	15
Iters	28	28	28	28
CPU time	18.315	26.526	36.235	34.971
SVD calls	23	20	18	13
SVD time	8.440	14.558	21.079	27.623
% SVD time	46.1	54.9	58.9	62.8

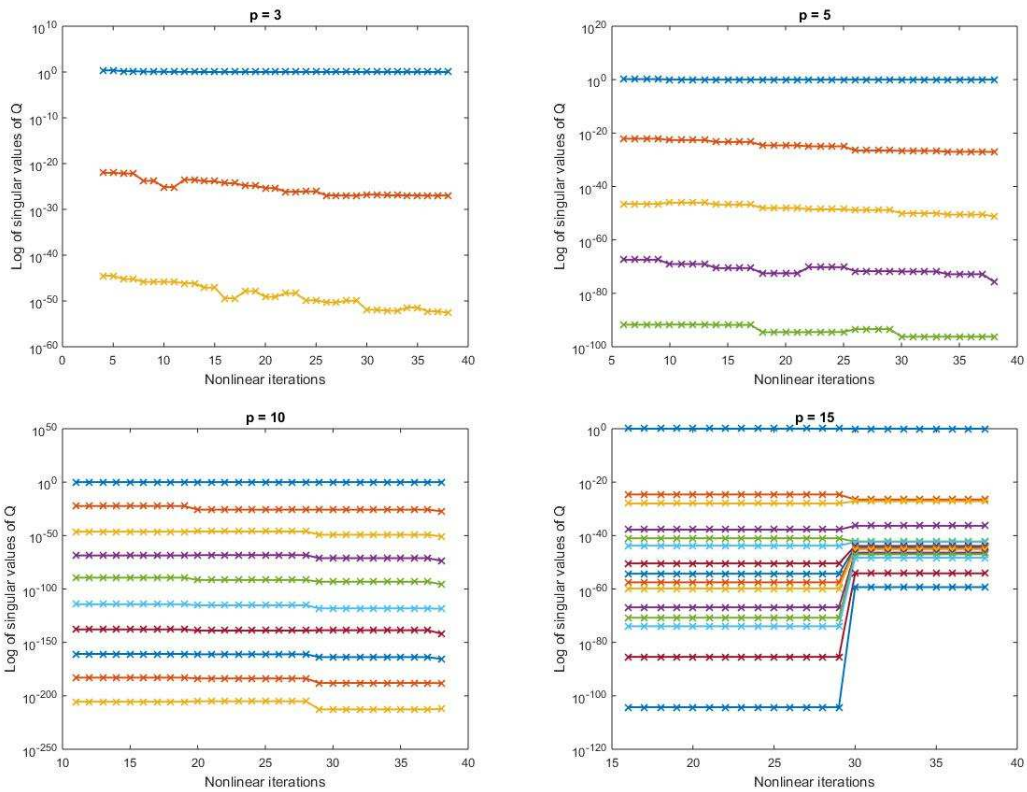

Figure 2. Singular values of the update matrix Q for $p = 3, 5, 10$ and $p = 15$ for Problem 2.

$$-\Delta u + Cu\left(\frac{\partial u}{\partial x} + \frac{\partial u}{\partial y}\right) = f \qquad (18)$$

with homogeneous Dirichlet boundary conditions on the unit square $(0,1) \times (0,1)$. As in [2], the source f has been constructed so that the exact solution is

$$10xy(1-x)(1-y)\exp(x^{4.5}).$$

Table 3. Performance of the DBRR method for Problem 2.

	$p=3$				$p=5$			
$\varepsilon =$	10^{-1}	10^{-3}	10^{-5}	10^{-10}	10^{-1}	10^{-3}	10^{-5}	10^{-10}
Iters	38	38	38	38	38	38	38	38
CPU time	10.588	10.566	10.647	10.839	11.434	11.333	11.606	11.265
SVD calls	18	18	18	18	9	9	9	9
SVD time	3.399	3.340	3.310	3.381	3.507	3.446	3.461	3.428
% SVD time	32.1	31.6	31.1	32.2	30.7	30.4	29.9	30.5
	$p=10$				$p=15$			
$\varepsilon =$	10^{-1}	10^{-3}	10^{-5}	10^{-10}	10^{-1}	10^{-3}	10^{-5}	10^{-10}
Iters	38	38	38	38	38	38	38	38
CPU time	14.705	14.778	14.795	14.693	17.930	19.642	19.851	17.810
SVD calls	4	4	4	4	2	2	2	2
SVD time	4.505	4.462	4.490	4.500	5.244	6.257	4.260	5.224
% SVD time	30.7	30.2	30.4	30.7	29.2	31.8	29.6	29.39

Table 4. Performance of the BRR method for Problem 2.

p	3	5	10	15
Iters	38	38	38	38
CPU time	14.531	22.864	46.849	78.311
SVD calls	35	33	28	23
SVD time	6.225	12.021	29.772	3
% SVD time	46.1	54.9	58.9	62.8

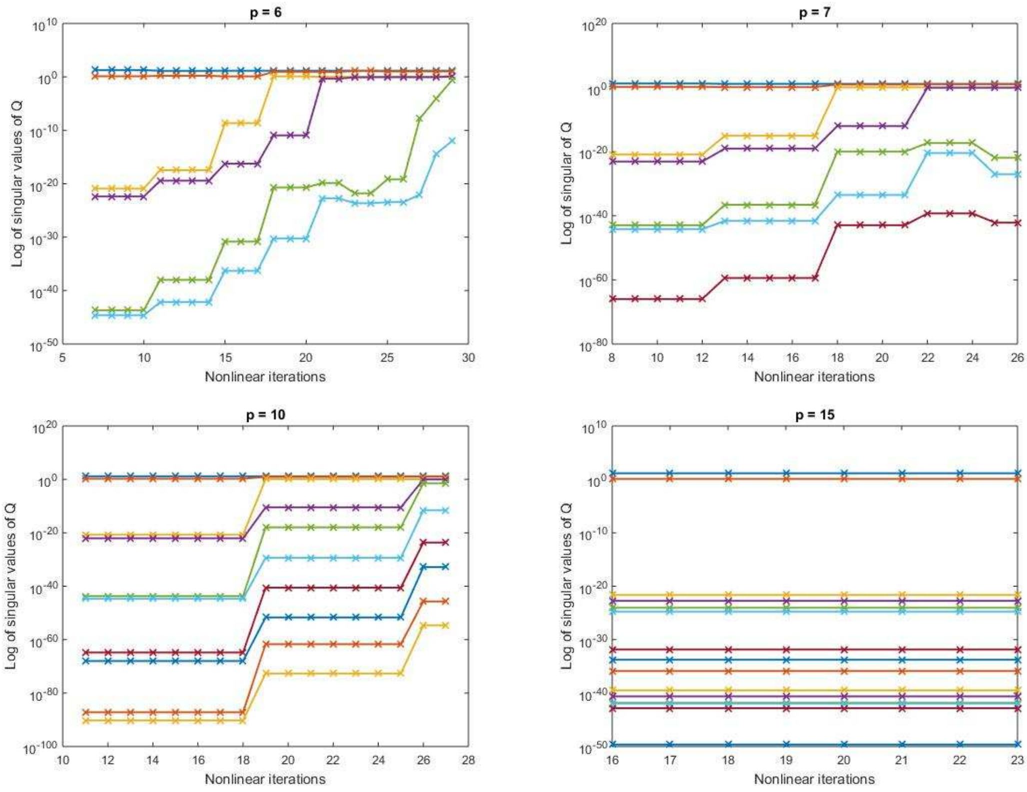

Figure 3. Singular values of the update matrix Q for $p = 6$, 7, 10 and $p = 15$ for Problem 3.

We set $C = 20$ and $u_0 = 0$. Equation (18) is discretized on a 500×500 grid using centered differences. Evolution of the sigular values of Q is displayed in figure 4 for different values of p.

For $p \leq 5$ the rank of Q is p, and only the p^{th} singular value is removed. In this case, BRR and DBRR methods behave similarly, see tables 7 and 8. For $p = 10$ the rank of Q is ten but condition (8) is satisfied in the first iterations for $\varepsilon = 10^{-1}$

Table 5. Performance of the DBRR method for Problem 3.

	$p=6$				$p=7$			
$\varepsilon =$	10^{-1}	10^{-3}	10^{-5}	10^{-10}	10^{-1}	10^{-3}	10^{-5}	10^{-10}
Iters	34	30	30	29	35	31	31	31
CPU time	64.624	59.187	48.712	62.041	62.158	60.539	53.024	56.152
SVD calls	7	10	10	10	6	7	7	8
SVD time	3.588	5.334	5.085	5.229	4.109	4.695	4.756	5.039
% SVD time	5.6	9.0	10.5	8.4	6.6	7.7	8.9	9.0
	$p=10$				$p=15$			
$\varepsilon =$	10^{-1}	10^{-3}	10^{-5}	10^{-10}	10^{-1}	10^{-3}	10^{-5}	10^{-10}
Iters	44	28	28	28	22	22	22	22
CPU time	88.144	50.620	57.985	47.654	39.766	47.879	48.934	40.140
SVD calls	5	3	3	3	1	1	1	1
SVD time	7.098	3.698	2.262	4.777	1.232	1.232	1.638	1.201
% SVD time	6.9	7.3	3.9	7.9	3.1	3.1	3.3	3

Table 6. Performance of the BRR method for Problem 3.

p	6	7	10	15
Iters	30	26	26	22
CPU time	66.512	64.398	74.920	55.834
SVD calls	24	19	16	17
SVD time	13.291	11.841	16.890	15.771
% SVD time	20.0	18.4	22.5	28.3

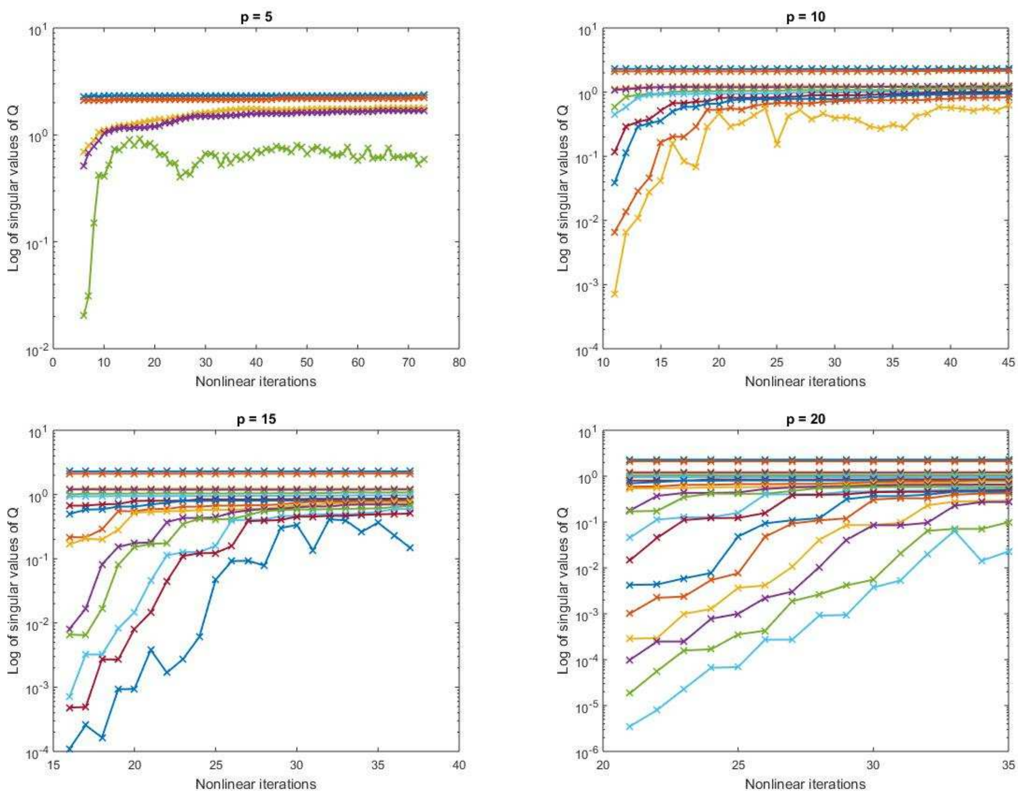

Figure 4. Singular values of the update matrix Q for $p = 5$, 10, 15 and $p = 20$ for Problem 4.

since, in this case, $q = 6$ at iteration 11 and $q = 8$ at both iterations 15 and 17. For $p = 20$ the rank of Q increases with nonlinear iterations, but condition (8) is satisfied for $\varepsilon = 10^{-1}, 10^{-3}, 10^{-5}$.

4. Conclusion

We have introduced a new Broyden rank reduction method to solve systems of nonlinear equations. The method is based on a thresholding

Table 7. Performance of the DBRR method for Problem 4.

	$p = 5$				$p = 10$			
$\varepsilon =$	10^{-1}	10^{-3}	10^{-5}	10^{-7}	10^{-1}	10^{-3}	10^{-5}	10^{-7}
Iters	73	73	73	73	45	45	45	45
CPU time	23.510	24.832	25.525	23.745	20.925	23.673	22.265	22.154
SVD calls	68	68	68	68	30	35	35	35
SVD time	5.945	6.283	6.406	5.960	7.608	9.508	9.083	8.976
% SVD time	25.3	25.7	25.1	25.1	36.8	40.1	40.8	40.5
	$p = 15$				$p = 20$			
$\varepsilon =$	10^{-1}	10^{-3}	10^{-5}	10^{-7}	10^{-1}	10^{-3}	10^{-5}	10^{-7}
Iters	36	38	37	37	36	35	35	35
CPU time	19.294	24.411	27.716	27.530	14.555	20.106	26.488	27.106
SVD calls	11	21	22	22	3	9	14	15
SVD time	6.797	10.992	13.067	12.661	2.979	8.078	12.978	13.575
% SVD time	35.4	45.1	47.2	46	20.5	40.2	49.0	50.1

Table 8. Performance of the BRR method for Problem 4.

p	5	10	15	20
Iters	73	45	37	35
CPU time	23.525	22.326	25.567	29.168
SVD calls	68	35	22	15
SVD time	5.825	9.002	12.045	14.646
% SVD time	24.8	41.0	47.1	50.9

process of the Broyden matrix singular values. All singular values of the update matrix that are smallest than a given threshold are removed. We have proved the local superlinear convergence and numerically tested the method on a variety of problems. As compared to classical Broyden rank reduction, our method induces a significant reduction of the execution time (CPU time) by reducing the number of calls of the singular values decomposition. As a perspective of this work is to combine the proposed method with that presented in [8].

Acknowledgements

This work has been partially supported by Hydrinv Euro-Mediterranean Project, LERMA, LIRIMA, IMIST and CNRST.

References

[1] Dennis, J. E. and Schnabel, R. B. (1983). Numerical methods for unconstrained optimization and nonlinear equations. Prentice-Hall.

[2] Kelley, C. T. (1995). Iterative methods for linear and nonlinear equations. Volume 16 of Frontiers in Applied Mathematics, SIAM, Philadelphia.

[3] Ortega, J. M. and Reinboldt, W. C. (2000). Iterative solution of nonlinear equations in several variables. SIAM, Philadelphia.

[4] Broyden, C. G. (1965). A class of methods for solving nonlinear simultaneous equations. *Mathematics of Computation*, (19), 577–593.

[5] Broyden, C. G., Dennis, J. E. and Moré, J. J. (1973). On the local and superlinear convergence of quasi-Newton methods. *IMA Journal of Applied Mathematics*, (12), 223–245.

[6] Byrd, R. H., Nocedal, J. and Schnabel, R. B. (1994). Representations of quasi-Newton matrices and their use in limited memory methods. *Mathematical Programming*, (63), 129–156.

[7] Van De Rotten, B. and Verduyn Lunel, S. M. (2003). A limited memory Broyden method to solve high dimensional systems of nonlinear equations. Technical Report MI 2003-06, Mathematical Institute, University of Leiden, The Netherlands.

[8] Ziani M. and Guyomarc'h F. (2008). An autoadaptative limited memory Broyden's method to solve systems of nonlinear equations. *Applied Mathematics and Computation*, (205), 202–211.

Mostafa Ouarit *is currently an Associate Professor and member of the Laboratory of Modelling, Analysis, Control and statistics at the Faculty of Sciences Ain Chock, Hassan II University Of Casablanca, Morocco. He received the PhD degree in Applied Mathematics at University of Technology of Compiégne, France. His research interests are in the areas of numerical analysis, optimization and optimal control.*

Ali Souissi *is Professor at the faculty of sciences, Mohammed V University, Rabat, Morocco. His main areas of interest are Scientific Computing, Optimal shape Design and Numerical Analysis of Partial Differential Equations. He received the PhD in Applied Mathematics from Laval University, Canada. He has publications in well-known journals and supervised many PhD theses.*

Mohammed Ziani *received his PhD in applied Mathematics at the University of Rennes (France) and Mohammed V University in Rabat (Morocco). He is professor of applied Mathematics at Faculty of sciences in Rabat. His research interests include issues related to solving large nonlinear systems of equations and partial differential equations based image processing.*

An International Journal of Optimization and Control: Theories & Applications (http://ijocta.balikesir.edu.tr)