Available online at www.sciencedirect.com
SCIENCE DIRECT®
ELSEVIER
Journal of Computational and Applied Mathematics 196 (2006) 478–484
JOURNAL OF COMPUTATIONAL AND APPLIED MATHEMATICS
www.elsevier.com/locate/cam

Spectral gradient projection method for solving nonlinear monotone equations

Li Zhang*, Weijun Zhou

Department of Applied Mathematics, Hunan University, Changsha 410082, China

Received 6 June 2005; received in revised form 30 September 2005

Abstract

An algorithm for solving nonlinear monotone equations is proposed, which combines a modified spectral gradient method and projection method. This method is shown to be globally convergent to a solution of the system if the nonlinear equations to be solved is monotone and Lipschitz continuous. An attractive property of the proposed method is that it can be applied to solving nonsmooth equations. We also give some preliminary numerical results to show the efficiency of the proposed method.

MSC: 90C25; 90C33

Keywords: Spectral gradient method; Monotone; Projection method

1. Introduction

In this paper, we consider the problem to find solutions of the following nonlinear equations

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

where $F : R^n \to R^n$ is continuous and monotone, i.e. $\langle F(x) - F(y), x - y \rangle \geqslant 0$ for all $x, y \in R^n$. Nonlinear monotone equations arise in various applications such as subproblems in the generalized proximal algorithms with Bregman distances [10]. Some monotone variational inequality can be converted into systems of nonlinear monotone equations by means of fixed point map or normal map [20].

For solving smooth systems of equations, the Newton method, quasi-Newton methods, Levenberg–Marquardt method and their variants are of particular importance because of their locally fast convergent rates [5–7,15]. To ensure global convergence, some line search strategy for some merit function are used, see [11]. Recently, Solodov and Svaiter [19] presented a Newton-type algorithm for solving systems of monotone equations. By using hybrid projection method, they showed that their method converges globally. For nonlinear equations, Griewank [9] obtained a global convergence results for Broyden's rank one method. Li and Fukushima [12] presented a Gauss–Newton-based BFGS method for solving symmetric nonlinear equations and established global convergence of their method. But these methods need to compute and store matrix which is not suitable for large scale nonlinear equations. To overcome this drawback, Nocedal

* Corresponding author. Tel.: +86 0731 8828 171; fax: +86 0731 8823 826.
E-mail addresses: zl606@tom.com (L. Zhang), weijunzhou@126.com (W. Zhou).

0377-0427/$ - see front matter
doi:10.1016/j.cam.2005.10.002

[14] proposed the limited memory BFGS method (L-BFGS) for unconstrained minimization problems. Numerical results [13,3] showed that the L-BFGS method is very competitive due to its low storage.

For solving nonsmooth systems of equations, many algorithms are proposed such as semismooth Newton method [16], other methods can be found in [8] and its references.

Recently, Cruz and Raydan [4] extended the spectral gradient method to solve nonlinear equations. Spectral gradient method are low cost nonmonotone schemes for finding local minimizers. It was introduced by Barzilai and Borwein [1]. The convergence for quadratics was established by Raydan [17] and a global scheme was discussed more recently for nonquadratic functions by Raydan [18]. It has been applied successfully to solving large scale unconstrained optimization problems [4,2]. In [4], Cruz and Raydan still converted the nonlinear equations into a unconstrained minimization problem by using some merit function.

In this paper, we propose a method to solve (1.1), which combines spectral gradient method [1] and projection method [19]. This method is very different from that of [4] because we do not use any merit function and any descent method. Under mild assumptions, we prove this method globally converges to solutions of (1.1). In Section 2, we state our algorithm. In Section 3, we establish the global convergence of the method. In Section 4, we give some numerical results.

2. Algorithm

In this section, we first introduce the spectral gradient method for unconstrained minimization problem:

$$\min f(x) \quad x \in R^n, \tag{2.1}$$

where $f : R^n \to R$ is continuously differentiable and its gradient is available. Spectral gradient method is defined by

$$x_{k+1} = x_k - \lambda_k g_k, \tag{2.2}$$

where g_k is the gradient vector of f at x_k and λ_k is given by

$$\lambda_k = \frac{s_{k-1}^{\mathrm{T}} s_{k-1}}{u_{k-1}^{\mathrm{T}} s_{k-1}}, \tag{2.3}$$

where $s_{k-1} = x_k - x_{k-1}$, $u_{k-1} = g_k - g_{k-1}$.

About projection method [19], note that by monotonicity of F, for any $\bar{x}$ such that $F(\bar{x}) = 0$, we have

$$\langle F(z_k), \bar{x} - z_k \rangle \leqslant 0. \tag{2.4}$$

By performing some kind of line search procedure along the direction d_k, a point $z_k = x_k + \alpha_k d_k$ computed such that

$$\langle F(z_k), x_k - z_k \rangle > 0. \tag{2.5}$$

Thus, the hyperplane

$$\mathscr{H}_k = \{x \in R^n \mid \langle F(z_k), x - z_k \rangle = 0\} \tag{2.6}$$

strictly separates the current iterate x_k from zeros of the systems of equations. Once the separating hyperplane is obtained, the next iterate x_{k+1} is computed by projecting x_k onto it.

We now can state our algorithm for solving (1.1).

Algorithm 1. *Step* 0: Choose an initial point $x_0 \in R^n$, $\beta \in (0, 1)$, $\sigma \in (0, 1)$, $r > 0$. Let $k := 0$.

Step 1: Compute d_k by

$$d_k = \begin{cases} -F(x_k) & \text{if } k = 0, \\ -\theta_k F(x_k) & \text{otherwise}, \end{cases} \tag{2.7}$$

where $\theta_k = s_{k-1}^{\mathrm{T}} s_{k-1} / y_{k-1}^{\mathrm{T}} s_{k-1}$ is similar to (2.3), $s_{k-1} = x_k - x_{k-1}$, but y_{k-1} is defined by

$$y_{k-1} = F(x_k) - F(x_{k-1}) + r s_{k-1},$$

which is different from the standard definition of y_{k-1}.

Stop if $d_k = 0$; otherwise,

Step 2: Determine steplength α_k, find $z_k = x_k + \alpha_k d_k$, where $\alpha_k = \beta^{m_k}$ with m_k being the smallest nonnegative integer m such that

$$-\langle F(x_k + \beta^m d_k), d_k\rangle \geqslant \sigma\beta^m \|d_k\|^2. \tag{2.8}$$

Step 3: Compute

$$x_{k+1} = x_k - \frac{\langle F(z_k), x_k - z_k\rangle}{\|F(z_k)\|^2} F(z_k). \tag{2.9}$$

Set $k := k + 1$ and go to step 1.

Remarks. (i) In step 1, by the monotonicity of F, we have

$$y_{k-1}^{\mathrm{T}} s_{k-1} = \langle F(x_k) - F(x_{k-1}), x_k - x_{k-1}\rangle + r s_{k-1}^{\mathrm{T}} s_{k-1} \geqslant r s_{k-1}^{\mathrm{T}} s_{k-1} > 0. \tag{2.10}$$

In addition, if F is Lipschitz continuous, i.e., there exists a constant $L > 0$ such that

$$\|F(x) - F(y)\| \leqslant L\|x - y\| \quad \forall x, y \in R^n. \tag{2.11}$$

Then we also have

$$y_{k-1}^{\mathrm{T}} s_{k-1} = \langle F(x_k) - F(x_{k-1}), x_k - x_{k-1}\rangle + r s_{k-1}^{\mathrm{T}} s_{k-1} \leqslant (L + r) s_{k-1}^{\mathrm{T}} s_{k-1}. \tag{2.12}$$

So we have from (2.7), (2.10) and (2.12)

$$\frac{\|F(x_k)\|}{L + r} \leqslant \|d_k\| \leqslant \frac{\|F(x_k)\|}{r}. \tag{2.13}$$

(ii) From (i) and (2.8), the step 2 is well-defined and so is the Algorithm 1.

(iii) Line search (2.8) is different from that of [19].

3. Convergence property

In order to obtain global convergence, we need the following lemma.

Lemma 3.1 (*Solodov and Svaiter* [19]). *Let F be monotone and $x, y \in R^n$ satisfy $\langle F(y), x - y\rangle > 0$. Let*

$$x^+ = x - \frac{\langle F(y), x - y\rangle}{\|F(y)\|^2} F(y).$$

Then for any $\bar{x} \in R^n$ such that $F(\bar{x}) = 0$, it holds that

$$\|x^+ - \bar{x}\|^2 \leqslant \|x - \bar{x}\|^2 - \|x^+ - x\|^2.$$

Now we can state our convergence result whose proof is similar to that of [19].

Theorem 3.2. *Suppose that F is monotone and Lipschitz continuous and let $\{x_k\}$ be any sequence generated by Algorithm 1. We also suppose the solution set of the problem is nonempty. For any $\bar{x}$ such that $F(\bar{x}) = 0$, it holds that*

$$\|x_{k+1} - \bar{x}\|^2 \leqslant \|x_k - \bar{x}\|^2 - \|x_{k+1} - x_k\|^2.$$

In particular, $\{x_k\}$ is bounded. Furthermore, it holds that either $\{x_k\}$ is finite and the last iterate is a solution, or the sequence is infinite and $\lim_{k\to\infty} \|x_{k+1} - x_k\| = 0$. Moreover, $\{x_k\}$ converges to some $\bar{x}$ such that $F(\bar{x}) = 0$.

Proof. We first note that if the algorithm terminates at some iteration k, then $d_k = 0$ and we have $F(x_k) = 0$ from (2.13), so that x_k is a solution. From now on, we assume that $d_k \neq 0$ for all k, then an infinite $\{x_k\}$ is generated.

We have from (2.8)

$$\langle F(z_k), x_k - z_k\rangle = -\alpha_k \langle F(z_k), d_k\rangle \geqslant \sigma\alpha_k^2 \|d_k\|^2 > 0. \tag{3.1}$$

Let $\bar{x}$ be any point such that $F(\bar{x}) = 0$. By (2.9), (3.1) and Lemma 3.1, it follows that

$$\|x_{k+1} - \bar{x}\|^2 \leqslant \|x_k - \bar{x}\|^2 - \|x_{k+1} - x_k\|^2. \tag{3.2}$$

Hence the sequence $\{\|x_k - \bar{x}\|\}$ is decreasing and convergent. Therefore, the sequence $\{x_k\}$ is bounded, and also

$$\lim_{k\to\infty} \|x_{k+1} - x_k\| = 0. \tag{3.3}$$

By (2.13), it holds that $\{d_k\}$ is bounded and so is $\{z_k\}$. Now by continuity of F, there exists a constant $C > 0$ such that $\|F(z_k)\| \leqslant C$.

We obtain from (2.9) and (3.1)

$$\|x_{k+1} - x_k\| = \frac{\langle F(z_k), x_k - z_k\rangle}{\|F(z_k)\|} \geqslant \frac{\sigma}{C}\,\alpha_k^2 \|d_k\|^2.$$

From the above inequality and (3.3), we have

$$\lim_{k\to\infty} \alpha_k \|d_k\| = 0. \tag{3.4}$$

If $\liminf_{k\to\infty} \|d_k\| = 0$, from (2.13), we have $\liminf_{k\to\infty} \|F(x_k)\| = 0$. Continuity of F implies that the sequence $\{x_k\}$ has some accumulation point $\hat{x}$ such that $F(\hat{x}) = 0$. From (3.2), it holds that $\{\|x_k - \hat{x}\|\}$ converges, and since $\hat{x}$ is an accumulation point of $\{x_k\}$, it must hold that $\{x_k\}$ converges to $\hat{x}$.

If $\liminf_{k\to\infty} \|d_k\| > 0$, from (2.13), we have $\liminf_{k\to\infty} \|F(x_k)\| > 0$. By (3.4), it holds that

$$\lim_{k\to\infty} \alpha_k = 0. \tag{3.5}$$

We have from (2.8)

$$-\langle F(x_k + \beta^{m_k - 1} d_k), d_k\rangle < \sigma\beta^{m_k-1} \|d_k\|^2. \tag{3.6}$$

Since $\{x_k\}$, $\{d_k\}$ are bounded, we can choose a subsequence, let $k \to \infty$ in (3.6), we obtain

$$-\langle F(\hat{x}), \hat{d}\rangle \leqslant 0, \tag{3.7}$$

where $\hat{x}$, $\hat{d}$ are limits of corresponding subsequences. On the other hand, by (2.13) and already familiar argument,

$$-\langle F(\hat{x}), \hat{d}\rangle > 0. \tag{3.8}$$

(3.7) and (3.8) are a contradiction. Hence $\liminf_{k\to\infty} \|F(x_k)\| > 0$ is not possible.

This finishes the proof. □

4. Numerical results

In this section, We only gave the following two simple problems to test the efficiency of the proposed method.

Problem 1. $F : R^n \to R^n$,

$$F_i(x) = x_i - \sin|x_i|, \quad i = 1, 2, \ldots, n,$$

where $F(x) = (F_1(x), F_2(x), \ldots, F_n(x))^{\mathrm{T}}$, $x = (x_1, x_2, \ldots, x_n)^{\mathrm{T}}$.

Table 1
Test results for our method, INM method and SANE method

		Our method		INM method		SANE method	
ip	n	iter	time	iter	time	iter	time
x_1	500	84	0.672	238	5.219	13	0.328
x_2	500	68	0.531	33	0.719	14	0.344
x_3	500	23	0.203	9	0.203	11	0.312
x_4	500	24	0.203	233	5.204	15	0.375
x_5	500	20	0.156	8	0.172	7	0.187
x_6	500	25	0.203	30	0.641	15	0.375
average	500	40.6667	0.328	91.8333	2.0263	12.5	0.32017
x_1	1000	103	1.829	331	25.218	14	1.11
x_2	1000	86	1.468	42	3.188	15	1.172
x_3	1000	23	0.437	9	0.703	11	0.891
x_4	1000	24	0.422	326	24.828	15	1.188
x_5	1000	21	0.375	9	0.718	8	0.641
x_6	1000	26	0.453	40	3.047	15	1.187
average	1000	47.1667	0.83067	126.1667	9.617	13	1.0315
x_1	3000	141	12.75	563	362.657	15	9.985
x_2	3000	124	10.719	66	40.844	15	9.313
x_3	3000	23	2.094	9	5.765	11	6.844
x_4	3000	25	2.235	558	347.125	16	9.938
x_5	3000	22	1.968	12	7.641	9	5.609
x_6	3000	27	2.359	63	39.25	16	9.954
average	3000	60.3333	5.3542	211.8333	133.8803	13.6667	8.6072

Problem 2. $F : R^n \to R^n$,

$$F_1(x) = 2x_1 + \sin(x_1) - 1,$$

$$F_i(x) = -2x_{i-1} + 2x_i + \sin(x_i) - 1, \quad i = 2, \ldots, n-1,$$

$$F_n(x) = 2x_n + \sin(x_n) - 1.$$

Obviously, Problem 1 is nonsmooth at $x = 0$.

We first compare our method with the Inexact Newton Method (INM) in [19] and the spectral approach for nonlinear equation (SANE) in [4]. We test Problem 1 for these three methods in Table 1 with different initial points $x_1 = (10, 10, \ldots, 10)^{\mathrm{T}}$, $x_2 = (1, 1, \ldots, 1)^{\mathrm{T}}$, $x_3 = (1, 1/2, \ldots, 1/n)^{\mathrm{T}}$, $x_4 = (-10, -10, \ldots, -10)^{\mathrm{T}}$, $x_5 = (-0.1, -0.1, \ldots, -0.1)^{\mathrm{T}}$, $x_6 = (-1, -1, \ldots, -1)^{\mathrm{T}}$. For our method, we set $\beta = 0.4$, $\sigma = 0.01$, $r = 0.001$. For INM method in [19], we set $\mu_k = \|F(x_k)\|$, $\rho_k = 0$, $\beta = 0.4$, $\lambda = 0.01$. For SANE method in [4], we set $\alpha_0 = 1$, $M = 1$, $\gamma = 0.01$, $\sigma = 0.4$, $\delta = 10$. We use the stopping criterion $\|F(x_k)\| < 10^{-4}$. The algorithms were coded in MATLAB7.0 and run on Personal Computer with 2.0 GHZ CPU processor. In Table 1, "n" is the dimension of the problem, "ip" means the initial point, "iter" stands for the total number of iterations, "time" stands for CPU time in second, "average" is the average of these numbers respectively.

From Table 1, we can see that the INM method perform worst, it needs most iterations and CPU time. Especially for initial points x_1, x_4, these two points are far from the solution of Problem 1, which affects the local fast convergence. Moreover, Problem 1 is not differentiable at $x = 0$, which cannot guarantee the superlinear convergence of the INM method. Our method need more iterations than the SANE method, but needs less CPU time especially for high dimension case which is important for large scale problems. So the results of Table 1 show our method is very efficient.

Now we discuss the role of the parameter r in our proposed method. Tables 2–3 show the number of iterations with different dimension sizes and different initial points when our method solve Problems 1 and 2, respectively. In Tables 2 and 3, the meaning of x_0 is the initial point. In Table 2, A/B stand for the total number of iterations in the

Table 2
Test results for our method on Problem 1 with $r = 0.1, 0.001$

x_0^T	$(0.1, \ldots, 0.1)$	$(1, \ldots, 1)$	$(1, 1/2, \ldots, 1/n)$	$(0, 0, \ldots, 0)$	$(-0.1, \ldots, -0.1)$	$(-1, \ldots, -1)$
$n = 100$	170/27	204/38	96/23	1/1	20/19	25/24
$n = 500$	309/58	343/68	96/23	1/1	21/20	27/25
$n = 1000$	396/77	431/86	96/23	1/1	22/21	27/26
$n = 2000$	506/99	540/109	96/23	1/1	23/22	28/27
$n = 3000$	583/114	617/124	96/23	1/1	23/22	29/27

Table 3
Test results for our method on Problem 2 with $r = 1, 0.1$

x_0^T	$(0.1, \ldots, 0.1)$	$(1, \ldots, 1)$	$(1, 1/2, \ldots, 1/n)$	$(0, 0, \ldots, 0)$	$(-0.1, \ldots, -0.1)$	$(-1, \ldots, -1)$
$n = 100$	379/313	376/269	375/280	378/292	380/290	385/295
$n = 500$	1549/1205	1538/1335	1541/1393	1530/1286	1540/1343	1547/1372
$n = 1000$	2801/2047	2794/2019	2790/2383	2802/2468	2786/2650	2797/2215
$n = 2000$	4385/4203	4382/3482	4375/3624	4385/3431	4388/3459	4380/3721
$n = 3000$	5328/3993	5325/3986	5338/3937	5329/4293	5333/5029	5320/3640

Algorithm 1 with $r = 0.1, 0.001$ respectively. In Table 3, A/B stand for the total number of iterations in the Algorithm 1 with $r = 1, 0.1$ respectively.

Tables 2 and 3 show that our method always stopped successfully for each problem. For Problem 1, the initial points influence the number of iterations very much. For Problem 2, numerical results indicate that the proposed method perform well, but the initial points do not influence the number of iterations very much. For high dimension case, the iteration number is large since we don't use enough derivative information of the equations. Moreover, the results show that the values of r affects significantly the performance of the algorithm. The smaller r is, the smaller the number of iteration is. This can explain partly as follows: from the Theorem 3.2, we have

$$\|x_{k+1} - \bar{x}\|^2 \leqslant \|x_k - \bar{x}\|^2 - \|x_{k+1} - x_k\|^2.$$

From (2.9) and definitions of z_k and d_k, we have

$$\begin{aligned}\|x_{k+1} - x_k\|^2 &= \frac{|\langle F(z_k), x_k - z_k\rangle|^2}{\|F(z_k)\|^2} \approx \|x_k - z_k\|^2 = \|\alpha_k d_k\|^2 \\ &= \alpha_k^2 \|F(x_k)\|^2 \theta_k^2 = \alpha_k^2 \|F(x_k)\|^2 \left(\frac{s_{k-1}^T s_{k-1}}{y_{k-1}^T s_{k-1}}\right)^2.\end{aligned}$$

From the above inequalities and the definition $y_{k-1} = F(x_k) - F(x_{k-1}) + r s_{k-1}$, we can get that when r get smaller, then x_k come closer to solutions of $F(x) = 0$.

Acknowledgements

The authors would like to thank the anonymous reviewers for their very helpful comments on the paper.

References

[1] J. Barzilai, J.M. Borwein, Two point stepsize gradient methods, IMA J. Numer. Anal. 8 (1988) 141–148.
[2] E.G. Birgin, Y.G. Evtushenko, Automatic differentiation and spectral projected gradient methods for optimal control problems, Optim. Meth. Soft. 10 (1998) 125–146.
[3] R.H. Byrd, J. Nocedal, C. Zhu, Towards a discrete Newton method with memory for large-scale optimization, Optimization Technology Center Report OTC-95-1, EEC Department, Northwestern University, 1995.

[4] W. Cruz, M. Raydan, Nonmonotone spectral methods for large-scale nonlinear systems, Optim. Meth. Soft. 18 (2003) 583–599.
[5] J.E. Dennis, J.J. Moré, A characterization of superlinear convergence and its application to quasi-Newton methods, Math. Comput. 28 (1974) 549–560.
[6] J.E. Dennis, J.J. Moré, Quasi-Newton method, motivation and theory, SIAM Rev. 19 (1977) 46–89.
[7] J.E. Dennis, R.B. Schnabel, Numerical methods for unconstrained optimization and nonlinear equations, Prentice-Hall, Englewood Cliffs, NJ, 1983.
[8] F. Facchinei, J.S. Pang, Finite-Dimensional Variational Inequalities and Complementarity Problems, Springer, New York, 2003.
[9] A. Griewank, The global convergence of Broyden-like methods with a suitable line search, J. Austral. Math. Soc. Ser. B 28 (1986) 75–92.
[10] A.N. Iusem, M.V. Solodov, Newton-type methods with generalized distances for constrained optimization, Optimization 41 (1997) 257–278.
[11] H. Jiang, D. Ralph, Global and local superlinear convergence analysis of Newton-type methods for semismooth equations with smooth least squares, Department of Mathematics, The University of Melbourne, Australia, July 1997.
[12] D.H. Li, M. Fukushima, A globally and superlinear convergent Gauss–Newton-based BFGS methods for symmetric nonlinear equations, SIAM J. Numer. Anal. 37 (1999) 152–172.
[13] D.C. Liu, J. Nocedal, On the limited memory BFGS method for large scale optimization methods, Math. Program. 45 (1989) 503–528.
[14] J. Nocedal, Updating quasi-Newton matrixes with limited storage, Math. Comput. 35 (1980) 773–782.
[15] J.M. Ortega, W.C. Rheinboldt, Iterative Solution of Nonlinear Equations in Several Variables, Academic Press, New York, 1970.
[16] L. Qi, J. Sun, A nonsmooth version of Newton's method, Math. Program. 58 (1993) 353–367.
[17] M. Raydan, On the Barzilai and Borwein choice of step length for the gradient method, IMA J. Numer. Anal. 13 (1993) 321–326.
[18] M. Raydan, The Barzilai and Borwein gradient method for the large scale unconstrained minimization problem, SIAM J. Optim. 7 (1997) 26–33.
[19] M.V. Solodov, B.F. Svaiter, A globally convergent inexact Newton method for systems of monotone equations, in: M. Fukushima, L. Qi (Eds.), Reformulation: Nonsmooth, Piecewise smooth, Semismooth and Smoothing Methods, Kluwer Academic Publishers, 1998, pp. 355–369.
[20] Y.B. Zhao, D. Li, Monotonicity of fixed point and normal mapping associated with variational inequality and its application, SIAM J. Optim. 4 (2001) 962–973.