广西科学 Guangxi Sciences 2006，13(4)：288～292

A Modified Gauss-Newton-based BFGS Method for Symmetric Nonlinear Equations*

一个修改的求解非线性对称方程组的高斯-牛顿 BFGS 方法

YUAN Gong-lin[1], WEI Zeng-xin[2], LU Xi-wen[1]

袁功林[1]，韦增欣[2]，鲁习文[1]

(1. School of Science, East China University of Science and Technology, Shanghai, 200237, China; 2. College of Mathematics and Information Science, Guangxi University, Nanning, Guangxi, 530004, China)

(1. 华东理工大学理学院，上海 200237；2. 广西大学数学与信息科学学院，广西南宁 530004)

Abstract: In this paper, a modified Gauss-Newton-based BFGS method based on the technique of Li and Fukushima [10] is proposed. The given method possesses the global and superlinear convergence under mild conditions. The presented method is better than the normal method for the given problem.

Key words: symmetric equations, BFGS method, global convergence, superlinear convergence

摘要：在文献[10]的基础上，给出一个修改的求解非线性对称方程组问题的高斯-牛顿 BFGS 方法，并建立该方法的全局和超线性收敛性. 该方法比原方法的效果要好.

关键词：对称方程组 BFGS 方法 全局收敛 超线性收敛

中图法分类号：O241.7 **文献标识码**：A **文章编号**：1005-9164(2006)04-0288-05

1 Introduction

It's well known that the BFGS method is a very effective method for solving optimization problems[1~5]. Some modified BFGS methods with global and superlinear convergence have been proposed[6~9]. Li and Fukushima[10] present a Gauss-Newton-based BFGS method for symmetric nonlinear equations, and get some better results. Motivated by their ideas, Wei et al[11] and Yuan et al[12,13] make a further study.

In this paper, we consider the following system of nonlinear equations

$$g(x)=0, x\in \mathscr{R}^n, \tag{1.1}$$

where $g: \mathscr{R}^n \rightarrow \mathscr{R}^n$ is continuously differentiable, and the Jacobian $\nabla g(x)$ of g is symmetric for all $x\in R^n$. Let θ be the norm function defined by $\theta(x)=\frac{1}{2}\|g(x)\|^2$. Then the nonlinear equation problem (1.1) is equivalent to the following global optimization problem

$$\min\theta(x), x\in \mathscr{R}^n. \tag{1.2}$$

For equation (1.1), Li and Fukushima[10] propose the following linear equation to get the search direction d_k

$$B_k d_k + \frac{g(x_k+\alpha_{k-1}g_k)-g_k}{\alpha_{k-1}} = 0, \tag{1.3}$$

where B_k is an approximation of matrix ∇g_k^2, g_k is the value of $g(x)$ at x_k (x_k is the kth iteration), and α_{k-1} is the steplength at the previous iteration. Matrix B_k is updated by the BFGS formula

$$B_{k+1} = B_k - \frac{B_k s_k s_k^T B_k}{s_k^T B_k s_k} + \frac{y_k y_k^T}{y_k^T s_k}, \tag{1.4}$$

where $s_k = x_{k+1} - x_k$, $y_k = g(x_k+\delta_k) - g(x_k)$, and $\delta_k = g_{k+1} - g_k$. Here y_k differs from the standard update formula where y_k is the difference of the gradients $g_{k+1} - g_k$, which is denoted by δ_k in this paper. The steplength α_k is generated by

$$\|g(x_k+\alpha d_k)\|^2 - \|g_k\|^2 \leqslant -\sigma_1\|\alpha g_k\|^2 - \sigma_2\|\alpha d_k\|^2$$

收稿日期：2006-04-14

修回日期：2006-06-23

作者简介：袁功林(1976-)，男，河南商丘人，助教，主要从事优化理论与方法的教学和研究工作。

* This Work is Supportted by Guangxi University SF Grands (X061041).

$+ \varepsilon_k \|g_k\|^2$, (1.5)

where σ_1 and σ_2 are some positive constants. $\{\varepsilon_k\}$ is a positive sequence satisfying

$$\sum_{k=0}^{\infty} \varepsilon_k < \infty. \quad (1.6)$$

The purpose of this paper is to present a modified Gauss-Newton-based BFGS method. The main difference from reference [10] is that: we use the following equations to get the d_k

$$B_k d_k + g_k = 0, \quad (1.7)$$

where B_k is generated by formula (1.4). The steplength α_k is generated by inequality (1.5) and

$$g(x_k + \alpha_k d_k)^T d_k \geqslant \sigma_3 g_k^T d_k, \quad (1.8)$$

where $\sigma_3 \in (0,1)$. The numerical results is very interesting comparing with Algorithm 1 in reference [10]. By Wolfe rule and the technique of reference [10], we can deduce that the search technique inequalities (1.5) and (1.8) are reasonable. Then the proposed method is well defined.

This paper is organized as follows. In the next section, the presented algorithm for solving equation (1.1) is stated. Under some reasonable conditions, the convergent results of the algorithm are established in Section 3. In Section 4, preliminary numerical results are reported.

2 The statement of algorithms

Algorithm 1

Step 0: Choose an initial point $x_0 \in R^n$, an initial symmetric positive definite matrix $B_0 \in R^{n\times n}$, a positive sequence $\{\varepsilon_k\}$ satisfying inequality (1.6), and constants $r, \rho, \sigma_3 \in (0,1), \sigma_1, \sigma_2 > 0, \alpha_{-1} > 0$, let $k = 0$;

Step 1: Stop if $\|g_k\| = 0$. Otherwise solve equation (1.7) to get d_k.

Step 2: If

$$\|g(x_k + d_k)\| \leqslant \rho \|g_k\|. \quad (2.1)$$

Then take $\lambda = 1$ and go to step 4. Otherwise go to step 3.

Step 3: Let i_k be the smallest nonnegative integer i such that inequalities (1.5) and (1.8) holds for $\alpha = r^i$. Let $\alpha_k = r^{i_k}$.

Step 4: Let the next iterative be $x_{k+1} = x_k + \alpha_k d_k$.

Step 5: Put $s_k = x_{k+1} - x_k = \alpha_k d_k$, $\delta_k = g_{k+1} - g_k$ and $y_k = g(x_k + \delta_k) - g(x_k)$. If $y_k^T s_k \leqslant 0$, then $B_{k+1} = B_k$ and go to step 6. Otherwise, update B_k by the BFGS formula (1.4).

Step 6: Let $k := k + 1$. Go to step 1.

Algorithm LF

In Algorithm 1, the step 1 and the step 3 are replaced by

Step 1: Stop if $\|g_k\| = 0$. Otherwise solve formula (1.3) to get d_k;

Step 3: Let i_k be the smallest nonnegative integer i such that inequality (1.5) holds for $\lambda = r^i$. Let $\lambda_k = r^{i_k}$.

Remark

(1) The step 5 of Algorithm 1 can ensure that B_k is always symmetric and positive definite, then equation (1.7) has a unique solution for each k. Moreover, for every k, step 3 can be executed in finite steps. Therefore, the method is well defined.

(2) Since $\{\varepsilon_k\}$ satisfies inequality (1.6), the inequalities (2.1) and (1.5) indicate that $\{g_k\}$ is at least approximately norm descent. Moreover, as we will see in Section 3, inequality (2.1) holds for all k sufficiently large. In other words, $\{g_k\}$ is norm descent when k issufficiently large.

3 Convergent analysis

Let Ω be the level set defined by

$$\Omega = \{x \mid \|g(x)\| \leqslant e^{\frac{\varepsilon}{2}} \|g(x_0)\|\}, \quad (3.1)$$

where ε is a positive constant such that

$$\sum_{k=0}^{\infty} \varepsilon_k \leqslant \varepsilon. \quad (3.2)$$

Lemma 3.1[10] Let $\{x_k\}$ be generated by Algorithm 1. Then $\{x_k\} \subset \Omega$. Moreover, $\{\|g_k\|\}$ converges.

In order to get the global convergence of Algorithm 1, the following Assumption is needed.

Assumption A

(i) g is continuously differentiable on an open convex set Ω_1 containing Ω.

(ii) The Jaconbian of g is symmetric and bounded on Ω_1 and there exists a positive constant M such that

$$\|\nabla g(x)\| \leqslant M \quad \forall x \in \Omega_1. \quad (3.3)$$

(iii) ∇g is uniformly nonsingular on Ω_1; i.e., there is a constant $m > 0$ such that

$$m\|d\| \leqslant \|\nabla g(x) d\| \quad \forall x \in \Omega_1, d \in R^n.$$

Remark Conditions (ii) in Assumption A implies that there exist constants $M \geqslant m > 0$ such that

$$m\|d\| \leqslant \|\nabla g(x) d\| \leqslant M\|d\| \forall x \in \Omega_1, d \in R^n, \quad (3.4)$$

$$m\|x - y\| \leqslant \|g(x) - g(y)\| \leqslant M\|x - y\| \forall x, y \in \Omega_1. \quad (3.5)$$

Under Assumption A, we can prove some useful properties pertaining to Algorithm 1.

Lemma 3.2 Let conditions (i) and (ii) in Assumption A be satisfied. Then the following inequalities hold for every k

$$\|\delta_k\| \leqslant M\|s_k\| \text{ and } \|y_k\| \leqslant M\|\delta_k\| \leqslant M^2\|s_k\|. \tag{3.6}$$

Proof Using inequality (3.5), we have

$\|\delta_k\| \leqslant M\|s_k\|$,

Now we prove the second inequality. By inequality (3.5) again, we get

$\|y_k\| \leqslant M\|\delta_k\| \leqslant M^2\|s_k\|$.

The proof is complete.

Lemma 3.3[10] Let Assumption A be satisfied. Then the following statements hold.

(i) If $s_k \to 0$, then there is a constant $m_1 > 0$ such that for all k sufficiently large

$$y_k^T s_k \geqslant m_1\|s_k\|^2. \tag{3.7}$$

(ii) Suppose that inequality (2.1) holds only for a finite number of k. Then we have

$$\sum_{k=0}^{\infty}\|\lambda_k g_k\|^2 < \infty \tag{3.8}$$

and

$$\sum_{k=0}^{\infty}\|\lambda_k d_k\|^2 = \sum_{k=0}^{\infty}\|s_k\|^2 < \infty. \tag{3.9}$$

Moreover, inequality (3.7) holds for all k sufficiently large.

Lemma 3.4 Let Assumption A hold. Then there are a positive integer k' and positive constants β_j, $j=1,2,3$, such that, for any $k \geqslant k'$, the inequalities

$$\beta_2\|s_i\|^2 \leqslant s_i^T B_i s_i \leqslant \beta_3\|s_i\|^2 \text{ and } \|B_i s_i\| \leqslant \beta_1\|s_i\| \tag{3.10}$$

hold for at least half of indices $i \in \{0,1,2,\cdots,k\}$.

Proof By Lemma 3.3, inequalities (3.6) and (3.7) hold for all k sufficiently large, say $k \geqslant k'$. From theorem 2.1 in reference [14], conditions inequalities (3.7) and (3.6) imply that $k' \leqslant i \leqslant k$. Since k' is a fixed integer and B_i are positive definite, we may take smaller β_2, and large β_1 and β_3 if necessary so that inequality (3.10) holds for all $i < k'$. Therefore inequality (3.10) holds for at least half of indices $i \in \{0,1,2,\cdots,k\}$.

Lemma 3.5 Let conditions (i) and (ii) in Assumption A hold. Then there exist constants $0 < m_0 \leqslant M_0$, we have the following estimate for α_k when k is large enough

$$\alpha_k \geqslant \frac{m_0}{M_0}. \tag{3.11}$$

Proof By inequality (1.8), we have

$$(g(x_k + \alpha_k d_k) - g_k)^T d_k \geqslant (\sigma_3 - 1)g_k^T d_k = -(1 - \sigma_3)g_k^T d_k. \tag{3.12}$$

Using $\|g(x_k + \alpha_k d_k) - g_k\|\|d_k\| \geqslant (g(x_k + \alpha_k d_k) - g_k)^T d_k$ and inequality (3.6), we get

$$M\alpha_k\|d_k\|^2 \geqslant \|g(x_k + \alpha_k d_k) - g_k\|\|d_k\| \geqslant -(1 - \sigma_3)g_k^T d_k. \tag{3.13}$$

On the other hand, using equation (1.7) and inequality (3.13), we obtain

$$M\alpha_k\|d_k\|^2 \geqslant (1 - \sigma_3)d_k^T B_k d_k. \tag{3.14}$$

Combining inequalities (3.10) and (3.14), we have

$$M\alpha_k\|d_k\|^2 \geqslant (1 - \sigma_3)d_k^T B_k d_k \geqslant (1 - \sigma_3)\beta_2\|d_k\|^2. \tag{3.15}$$

Then, we get $\alpha_k \geqslant \frac{\beta_2(1 - \sigma_3)}{M}$, let $m_0 = \beta_2(1 - \sigma_3)$ and $M_0 = M$. The proof is complete.

Now we establish a global convergence theorem for Algorithm 1.

Theorem 3.1 Let Assumption A hold. Then the sequence $\{x_k\}$ generated by Algorithm 1 converges to the unique solution x^* of equation (1.1).

Proof By Lemma 3.1, we know that $\{\|g_k\|\}$ is convergent. If

$$\liminf_{k\to\infty}\|g_k\| = 0, \tag{3.16}$$

then every accumulation point of $\{x_k\}$ is a solution of equation (1.1). Since $\nabla g(x)$ is uniformly nonsingular on Ω_1, equation (1.1) has only one solution. Moreover, since Ω is bounded, $\{x_k\} \in \Omega$ has at least one accumulation point. Therefore $\{x_k\}$ itself converges to the unique solution of equation (1.1). Thus it suffices to verify inequality (3.16).

If inequality (2.1) holds for infinitely many k's, then inequality (3.16) is trivial. Consider the case where inequality (2.1) holds for only finitely many k's, so that step 3 is executed for all k sufficiently large. Since inequality (3.8) holds, we need only to show that there is an infinite subsequence of $\{\alpha_k\}$ with a positive lower bounded, i. e.

$$\limsup_{k\to\infty}\alpha_k \geqslant 0.$$

Using inequality (3.11), it's obviously that the above formula is satisfied. The proof is complete.

Notice that theorem 3.1 ensures that $\{x_k\}$ converges. In particular, $s_k \to 0$. Therefore, Lemma 3.3 (i) yields that $y_k^T s_k > 0$ for all k sufficietly large. Hence we see from step 5 in Algorithm 1 that for all k large enough, B_{k+1} is always generated by the update formula (1.4).

Similar to the proof of theorem 3.9 in reference [10], it is not difficult to prove the superlinear result of Algorithm 1. Here we state the theorem as follows but omit the proof.

Theorem 3.2 Let Assumption A hold, and suppose that ∇g is Hölder continuous. Then the sequence $\{x_k\}$ that is generated by Algorithm 1 is superlinearly convergent.

4 Numerical results

In this section, we report results of some preliminary numerical experiments with the proposed method and Algorithm LF.

Problem. The discretized two-point boundary value problem [15]

$$g(x) = Ax + \frac{1}{(n+1)^2}F(x) = 0,$$

where A is the $n \times n$ tridiagonal matrix given by

$$A = \begin{bmatrix} 8 & -1 & & & \\ -1 & 8 & -1 & & \\ & -1 & 8 & -1 & \\ & & \ddots & \ddots & \ddots \\ & & & \ddots & \ddots & -1 \\ & & & & -1 & 8 \end{bmatrix},$$

and $F(x) = (F_1(x), F_2(x), \cdots, F_n(x))^T$ with $F_i(x) = \cos x_i - 1, i = 1, 2, \cdots, n$.

In the experiments, the parameters in Algorithm LF and Algorithm 1 were chosen as $r = 0.1, \rho = \sqrt{0.9}, \sigma_1 = \sigma_2 = 10^{-5}, \lambda_{-1} = 0.001, \sigma_3 = 0.95$, and $\varepsilon_k = k^{-2}$, where k is the number of iteration. The initial matrix B_0 was always set to be the unit matrix. The program was coded in MATLAB 7.0. We stopped the iteration when the condition $\|g(x)\| \leqslant 10^{-5}$ was satisfied. The tested results are listed in the following Tables 1～4. The columns of the tables have the following meaning:

Dim is the dimension of the problem.

NI is the total number of iterations.

NG is the number of the function evaluations.

For the given problem (Tables 1～4), we can see that the numerical results of the proposed method is more effectively than those of the Algorithm LF. The two methods have some common properties: the initial points do not influence the number of iterations very much, and the numerical results don't change obviously with the dimension increasingly.

Table 1 Test results for Algorithm LF

Dim	NI/NG					
	(1,…,1)	(50,…,50)	(500,…,500)	(−1,…,−1)	(−50,…,−50)	(−500,…,−500)
n=50	42/188	66/298	71/322	48/217	61/278	73/326
n=100	88/399	100/453	132/599	89/404	103/468	119/539
n=300	104/479	130/598	143/659	104/478	127/585	144/662
n=500	104/478	127/584	143/657	104/478	127/584	143/657

Table 2 Test results for Algorithm LF

Dim	NI/NG					
	(1,0,1,0…)	(50,0,50,0,…)	(500,0,500,0,…)	(−1,0,−1,0,…)	(−50,0,−50,0,…)	(−500,0,−500,0,…)
n=50	69/315	69/314	70/318	69/315	69/315	70/318
n=100	95/434	113/518	124/566	92/421	118/539	124/566
n=300	96/443	122/562	137/631	96/443	122/562	137/631
n=500	93/428	120/552	136/624	93/428	120/552	136/624

Table 3 Test results for Algorithm 1

Dim	NI/NG					
	(1,…,1)	(50,…,50)	(500,…,500)	(−1,…,−1)	(−50,…,−50)	(−500,…,−500)
n=50	62/155	76/192	102/244	60/149	90/223	102/244
n=100	65/168	86/221	92/235	65/168	76/193	89/227
n=300	64/160	75/188	88/221	63/157	75/188	85/214
n=500	66/165	80/200	88/221	72/180	83/208	93/232

Table 4 Test results for Algorithm 1

Dim	NI/NG					
	(1,0,1,0…)	(50,0,50,0,…)	(500,0,500,0,…)	(−1,0,−1,0,…)	(−50,0,−50,0,…)	(−500,0,−500,0,…)
n=50	53/137	69/177	86/213	53/137	67/173	85/210
n=100	56/143	78/198	81/207	55/139	69/174	76/193
n=300	56/140	70/176	84/211	57/143	68/171	81/204
n=500	56/143	70/178	80/204	56/143	70/178	80/204

References:

[1] DAI Y. Convergence properties of the BFGS algorithm [J]. SIAM Journal on Optimization, 2003, 3(13): 693-701.

[2] DENNIS J E, SCHNABEL R B. Numerical methods for unconstrained optimization and nonlinear equations[M]. NJ: Pretice-Hall Inc Englewood Cliffs, 1983.

[3] FLETCHER R. Practical methods of optimization[M]. 2nd ed. Chichester: John Wiley & Sons, 1987.

[4] GRIEWANK A, PH L. Toint, Local convergence analysis for partitioned quasi-Newton updates[J]. Number Math 1982, 39(3): 429-448.

[5] YUAN Y, SUN W. Theory and Methods of Optimization [M]. Beijing: Science Press of China, 1999.

[6] LI D, FUKUSHIMA M. A modified BFGS method and its global convergence in nonconvex minimization [J]. Journal of Computational and Applied Mathematics, 2001, 129(1): 15-35.

[7] POWELL M J D. A new algorithm for unconstrained optimation[M]//J B ROSEN, O L MANGASARIAN, K RITTER, eds. Nonlinear Programming. New York: Academic Press, 1970.

[8] WEI Z, QI L, CHEN X. An SQP-typemethod and its application in stochastic programming [J]. Journal of Optimization Theory and Applications, 2003, 116(1): 205-228.

[9] WEI Z, YU G, YUAN G, et al. The superlinear convergence of a modified BFGS-type method for unconstrained optimization [J]. Computational Optimization and Applications, 2004, 1(29): 315-332.

[10] LI D, FUKUSHIMA M. A globally and superlinearly convergent Gauss-Newton-based BFGS method for symmetric nonlinear equations [J]. SIAM Journal on Numerical Analysis, 1999, 37(1): 152-172.

[11] WEI Z, YUAN G, LIAN Z. An approximate Gauss-Newton-based BFGS method for solving symmetric nonlinear equations [J]. Guangxi Sciences, 2004, 11 (2): 91-99.

[12] YUAN G, LI X. An approximate Gauss-Newton-based BFGS method with descent directions for solving symmetric nonlinear equations [J]. OR Transactions, 2004, 8(4): 10-26.

[13] YUAN G, LU X. A nonmonotone Gauss-Newton-based BFGS method for solving symmetric nonlinear equations [J]. Journal of Lanzhou University: Natural Sciences, 2005, 41: 851-855.

[14] BYRD R, NOCEDAL J. A tool for the analysis of quasi-Newton methods with application to unconstrained minimization[J]. SIAM Journal on Numerical Analysis, 1989, 3(26): 727-739.

[15] MORé J J, GARBOW B S, HILLSTROME K E. Testing unconstrained optimization software[J]. ACM Trans Math Software, 1981, 7(1): 17-41.

(责任编辑:凌汉恩;英文编辑:蒋汉明)

(上接第287页 Continue from page 287)

[5] AIPING LIAO. Modifying BFGS method[J]. Operations Research Letters, 1997, 20: 171-177.

[6] LIU GUANGHUI, HAN JIYE, XU ZHONGLING. Global convergence of the variable metric algorithms with a generalized wolfe linesearch[J]. Journal of Mathematical Research and Exposition, 1995(4): 499-508.

[7] 韩继业,刘光辉.无约束最优化线搜索一般模型及BFGS方法的整体收敛性[J].应用数学学报,1995(1):112-122.

[8] YUAN YAXIANG, RICHARD, BYRD H. Non-quasi-Newton unconstrained optimization, IMA[J]. Journal of Computational Mathematics, 1995(2): 95-107.

[9] LI D, FUKUSHIMA M. Amodified BFGS method and its global convergence in nonconvex minimization [J]. J Comput Appl Math, 2001(129): 15-35.

[10] WEI Z, YU G, YUAN G, et al. The superlinear convergence of a modified BFGS type method for unconstrained optimization. Computational Optimization and Applications, 2004(25): 315-332.

[11] WEI Z, YU G. Some recent progress in unconstrained nonlinear optimization: proceedings of the 2003's International Conference on Numerical Optimization and Numerical Linear Algebra [C]. YUAN Y, eds. Beijing/Newyork: Science Press, 2004: 110-141.

[12] 袁亚湘,孙文瑜.最优化理论和方法[M].北京:科学出版社,1997.

[13] MORé J J, GARBOW B S, HILLSTROME K E. Testing unconstrained optimization software[J]. ACM Trans Math Software, 1981(7): 17-41.

(责任编辑:韦廷宗)