Iraqi Journal of Statistical Science (17) 2010
The Second Scientific Conference of Mathematics-Statistics& Information
pp[47 -54]

A Modified Curve Search Algorithm for Solving Unconstrained Optimization Problems

Abbas Y. AL-Bayati[*] **Bara'a M. Sulaiman**[**] **Suzan Y. Rassam**[***]

Abstract

In this paper, we present a modified algorithm with curve search rule for unconstrained minimization problems. At each iteration, the next iterative point is determined by means of a curve search rule. That the search direction and the step-size are particularly determined simultaneously at each iteration of the new algorithm.

خوارزمية منحني البحث المطورة لحل مسائل الأمثلية غير المقيدة

الملخص

في هذا البحث سنقدم خوارزمية مطورة مع قاعدة منحني البحث في مسائل الأمثلية غير المقيدة. تـم إيجاد النقطة التكرارية التالية عند كل تكرار باستخدام قاعدة منحني البحث. وبصورة خاصـة سـيتم إيجاد اتجاه البحث وحجم الخطوة في آن واحد عند كل تكرار في الخوارزمية الجديدة.

[*] Prof./ Department of Mathematics/College of Computer Sciences and Mathematics

[**] Ass·Lecture/Department of Mathematics/College of Computer Sciences and Mathematics

[***] Ass·Lecture /Department of Mathematics/College of Computer Sciences and Mathematics

Received:1/7 /2009 ____________________Accepted: 6 /12 / 2009

1. Introduction

Consider an unconstrained minimization problem (UP)

$$\min f(x), \quad x \in R^n \tag{1}$$

where $f : R^n \to R$ is a continuously differentiable function. Most of the well-known iterative algorithms for solving (UP) take the form

$$x_k = x_{k-1} + \alpha_k d_{k-1} \tag{2}$$

where d_k is a search direction, and α_k is a positive step-size parameter. If x_k is the current iterative point, then we denote $\nabla f(x_k)$by g_k, $f(x_k)$ by f_k and $f(x^*)$ by f^*, respectively.

If $d_k = -g_k$, then the corresponding method is called Steepest method. This method has low convergence rate in many situations, and often yields zigzag phenomenon. However, it does not require computing and strong some matrices associated with Hessian of objective functions.

If we take $d_k = -H_k g_k$ in (2), where the H_k is a matrix that approximates the inverse of the Hessian of f at x_k, the related methods are called Newton-like methods. It needs to store and compute the matrix associated with Hessian of f, but it has faster convergence rate than steepest descent method and conjugate gradient methods in many situations.

Generally, the conjugate gradient method is a useful technique for solving large scale minimization problems because it avoids, like steepest descent method, the computation and storage of some matrices. The conjugate gradient method has the form

$$d_k = \begin{cases} -g_k & \text{for k} = 1; \\ -g_k + \beta_k d_{k-1} & \text{for k} \geq 2, \end{cases} \tag{3}$$

where β_k is a parameter that defines the different conjugate gradient methods. However, many conjugate gradient methods have no global convergence e.g., (Bertsekas, 1982), (Evtushenko, 1985), (Grippo and Lucidi, 1997), (Hestens, 1980), (Nocedal, 1999), (Powell, 1977) and (Powell, 1976).

Miele and Cantrell (Miele and Cantrell, 1969) studied memory gradient method for (UP). Cantrell (Cantrall, 1969) showed that the memory gradient method and the Fletcher–Reeves algorithm (Fletcher and Reeves, 1964) were identical in the particular case of a quadratic function.

Cragg and Levy proposed a super-memory gradient method which is a generalization of Miele and Cantrell's method. (Cragg and Levy, 1969)

Wolfe and Viazminsky (Wolfe and Viazminsky, 1976) investigated a super-memory descent method for (UP). Other literatures on super-memory gradient method can be seen in e.g., (Qui and Shi, 2000), (Shi, 2003) and (Shi, 2000).

Both memory gradient method and super-memory gradient method are more efficient than conjugate gradient methods e.g., (Grippo and Lucidi, 1997), (Gilbert and Nocedal, 1992), (Powell, 1977) and (Powell, 1976) because they use more previous iterative information and add freedom of choosing parameters.

However, the convergence results of memory gradient methods and super memory gradient methods for non-quadratic objective functions are barely seen in recent literatures. It is of significance to investigate an efficient convergent super-memory gradient algorithm for solving large scale minimization problems, especially the problems in which the objective function is non-quadratic or even non-convex. As we know, the ODE methods (or dynamic methods) for unconstrained minimization are curve search methods (Botsaris, 1978), (Schropp, 1997), (Syman, 1982), (Van Wyk, 1984) and (Wu, Xia and Ouyang, 2002). It is required to solve some ordinary differential equations so as to approximate the minimizer of (UP). Ford et al. (Ford and Tharmlikit, 2003), (Ford and Moghrabi, 1996a) and (Ford and Moghrabi, 1996b) studied a new class of multi-step quasi-Newton methods for unconstrained minimization. However, it is required to store some matrices at each iteration. These methods are suitable to solve small and intermediate problems.

To accelerate the convergence rate and avoid the evaluation and storage of matrices, we present a new descent algorithm and prove its global convergence and linear convergence rate. At each iteration, the next iterative point is determined by means of a curve search rule that resembles Wolfe's line search rule. The algorithm, similarly to conjugate gradient methods, avoids the computation and storage of some matrices associated with the Hessian of objective functions. Though the algorithm in the paper has no as fast convergence rate as Newton-like methods, it is suitable to solve large scale minimization problems. The new algorithm is similar to Cragg and Levy's algorithm (Cragg and Levy, 1969), but is superior to it in the aspect of convergence. The algorithm is not a line search method, we may call it a curve search method.

2. New Algorithm

We assume that

(H_1): The function f has lower bound on $\Gamma = \{x \in R^n : f(x) \leqslant f(x_0)\}$, where x_0 is available.

(H_2): The gradient g is Lipschitz continuous in an open convex set B that contains L_0, i.e., there exists L > 0 such that

$$\|g(x) - g(y)\| \leqslant L \, \|x - y\|, \quad \forall x, y \in N \tag{4}$$

There are a lot of rules for choosing step-size ak e.g., (Cohen, 1981) and (Vrahatis, 2000) etc. We use curve search rule, which is similar to Wolfe's line search rule.

Curve search rule: At each iteration, fixed $s_k > 0$, the step-size α_k satisfies

$$f\left(x_k + \alpha d_k(\alpha)\right) - f_k \le \mu_1 \alpha_k g_k^T d_k \tag{5}$$

$$g\left(x_k + \alpha_k d_k(\alpha)\right)^T d_k(\alpha) \geqslant \mu_k g_k^T d_k \tag{6}$$

where $0 < \mu_1 < \frac{1}{2} < \mu_2 < 1$, and

$$d_k(\alpha) = -\delta_k \left(1 + \frac{\alpha s_k}{1+\alpha}\right) g_k - \frac{\alpha s_k}{1+\alpha} d_{k-1},$$

where

$$\delta_k = \frac{d_k^T y_{k-1}}{\|g_{k-1}\|^2}, \tag{7}$$

It is clear that, if ELS is used, then $g_k^T d_{k-1} = 0$. In this case we have

$$\delta_k = \frac{g_k^T d_{k-1} - d_{k-1}^T g_{k-1}}{\|g_{k-1}\|^2} = -\frac{d_{k-1}^T g_{k-1}}{\|g_{k-1}\|^2} = 1$$

For simplicity, we sometimes denote $d_k(\alpha_k)$ by d_k. It is obvious that the above search rule is not a line search rule though it is similar to Wolfe's line search rule. We may call it a modified curve search rule, in which the search direction and step size are determined at the same time. It is different from the traditional line search methods in which one first defines a descent direction and then finds a step-size along the direction. In the curve search method, search direction is a vector function of step-size. In fact, at the *k*th iteration, we find a new iterative point x_{k+1} along the curve $\{x_k + \alpha d_k(\alpha\ ) : \alpha \ge 0\}$ from the current point x_k.

Algorithm. $0 < \mu_1 < \frac{1}{2} < \mu_2 < 1$, $x_1 \in R^n$, $k = 1$, $\varepsilon > 0$

Step 1: If $\|g_k\| < \varepsilon$ then stop, else go to step 2;

Step 2: Define

$$s_k = \begin{cases} 1, & \textit{if } k = 1; \\ \|g_k\|^2 / \left[\|g_k\|^2 + \delta_k^{-1} \left|g_k^T d_{k\ \ 1}\right|\right], & \text{if k} \geqslant 2, \end{cases} \tag{8}$$

where $\delta_k = \frac{d_k^T y_{k-1}}{\|g_{k-1}\|^2}$

Step 3: $x_{k+1} = x_k + \alpha d_k(\alpha_k)$ where

$$d_k(\alpha_k) = \begin{cases} -g_k, & if\ k = 1; \\ -\left[\delta_k\left(1 - \frac{\alpha s_k}{1+\alpha}\right)g_k + \frac{\alpha s_k}{1+\alpha} d_{k\ 1}\right], & \text{if k } \geqslant 2, \end{cases} \tag{9}$$

and α_k is chosen by modified curve search rule;

Step 4: $k = k + 1$, go to step 1.

Traditional line search methods consist of two stages. The first one is to find a descent direction and the second is to define a step-size along the search direction. In the above algorithm, the search direction and the step-size are determined at the same time at each iteration and the search is along a curve.

With respect to the above algorithm, the first problem is whether α_k exists at each iteration. To solve this problem, we have the following result.

Lemma 2.1. Suppose that (H$_1$) holds. Let $\mu_1 \in \left(0, \frac{1}{2}\right)$, $\mu_2 \in (\mu_2, 1)$, and assume that $\|g_k\| \neq 0$, thus $s_k \neq 0$. There exists an interval $[c_1, c_2]$ with $0 < c_1 < c_2$, such that every $\alpha[c_1, c_2]$ satisfies (5) and (6).

The prove is easy to obtained.

Lemma 2.2. For all $k \geq 0$

$$-g_k^T d_k \geq \delta_k \frac{\|g_k\|^2}{1+\alpha_k}$$

Proof.

$$\begin{aligned} g_k^T d_k(\alpha_k) &= \delta_k\left(1 + \frac{\alpha_k s_k}{1+\alpha_k}\|g_k\|^2\right) + \frac{\alpha_k s_k}{1+\alpha_k} g_k^T d_{k\ 1} \\ &= \delta_k\|g_k\|^2 - \delta_k \frac{\alpha_k s_k}{1+\alpha_k}\left(\|g_k\|^2 - \delta_k^{-1}\left|g_k^T d_{k\ 1}\right|\right) \\ &\geqslant \delta_k\|g_k\|^2 - \delta_k \frac{\alpha_k s_k}{1+\alpha_k}\left(\|g_k\|^2 + \delta_k^{-1}\left|g_k^T d_{k\ 1}\right|\right) \\ &= \delta_k\|g_k\|^2 - \frac{\delta_k \alpha_k}{1+\alpha_k}\|g_k\|^2 = \frac{\delta_k}{1+\alpha_k}\|g_k\|^2 \end{aligned}$$

Lemma 2.3. If (H_2) holds, then

$$\alpha_k \geq -\frac{(1-\mu_2)g_k^T d_k(\alpha_k)}{L\|d_k(\alpha_k)\|^2}$$

Proof.

By (H_2), Cauchy-Schwarz inequality and (6), we have

$$\begin{aligned}\alpha_k L\|d_k(\alpha_k)\|^2 &\geq \|g(x_k+\alpha_k\theta_k d_k(\alpha_k))-g_k\|^2.\|d_k(\alpha_k)\| \\ &\geq [g(x_k+\alpha_k\theta_k d_k(\alpha_k))-g_k]^T d_k(\alpha_k) \\ &\geq -(1-\mu_2)g_k^T d_k(\alpha_k)\end{aligned}$$

i.e.,

$$\alpha_k \geq -\frac{(1-\mu_2)g_k^T d_k(\alpha_k)}{L\|d_k(\alpha_k)\|^2}$$

Numerical Result

We tested the FR method, PR method and New method. The test problems are drawn from (Andrie, 2004). The Numerical results of our tests are reported in Table 2.1. Each problem was tested with three different values of n ranging from n=100 to n=1000. The numerical results are given in the form of NOI / NOF, where NOI, NOF denote the number of iteration and function evaluations, respectively. The stopping condition is $\|g_k\| \leq 1\times 10^{-5}(1+|f_k|)$. From Table 2.1, we see that for these problems, NEW method really performs much better than the FR method and PR method.

Table 2.1. Numerical Results of FR method, PR method and NEW method

		FR Method	PR Method	NEW Method
Test Function	**n**	**NOI / NOF**	**NOI / NOF**	**NOI / NOF**
DIXMAANA	100	6 / 19	7 / 21	7 / 14
	500	6 / 21	6 / 21	7 / 14
	1000	8 / 24	6 / 22	7 / 14
Extended	100	Failed	Failed	22 / 41
Trigonometric	500	Failed	Failed	84 / 136
	1000	Failed	Failed	35 / 63
ARGLINB	100	Failed	1 / 6	1 / 3
	500	1 / 4	1 / 5	1 / 3
	1000	Failed	Failed	1 / 3
Raydan 2	100	2 / 8	2 / 8	4 / 9
	500	2 / 7	2 / 7	4 / 9
	1000	2 / 10	1 / 6	4 / 9
Full Hessian	100	305 / 677	308 / 724	3 / 7
	500	636 / 1301	1044 / 1638	3 / 7
	1000	934 / 1555	3150 / 3774	3 / 7

FLETCHER	100	863 / 1376	300 / 621	38 / 78
	500	4966 / 5468	1488 / 2072	28 /54
	1000	4984 / 5486	2984 / 3568	30 / 57

From this table and for six standard test functions with 18 problem-dimension cases, the NEW algorithm is superior on the standard well-known CG-methods for all the selected cases.

References

[1] N. Andrei, Unconstrained Optimization Test Functions, Research Institute for Informatics, 2004

[2] D.P. Bertsekas, Constrained Optimization and Lagrange Multiplier Methods, Academic Press Inc., 1982.

[3] C.A. Botsaris, Differential gradient methods, J. Math. Anal. Appl. 63 (1978) 177–198.

[4] A.I. Cohen, Step-size analysis for descent methods, JOTA 33 (2) (1981) 187–205.

[5] E.E. Cragg, A.V. Levy, Study on a super memory gradient method for the minimization of functions, JOTA 4 (3) (1969) 191–205.

[6] J.W. Cantrell, On the relation between the memory gradient and the Fletcher–Reeves method, JOTA 4 (1) (1969) 67–71.

[7] Y.G. Evtushenko, Numerical Optimization Techniques, Optimization Software Inc., Publications Division, New York, 1985.

[8] J.A. Ford, S. Tharmlikit, New implicit updates in multi-step quasi-Newton methods for unconstrained optimization, J. Comput. Appl. Math. 152 (2003) 133–146.

[9] J.A. Ford, I.A. Moghrabi, Using function-values in multi-step quasi-Newton methods, J. Comput. Appl. Math. 66 (1996) 201–211.

[10] J.A. Ford, I.A. Moghrabi, Minimum curvature multi step quasi-Newton methods, Comput. Math. Appl. 31 (4/5) (1996) 179–186.

[11] R. Fletcher, C. Reeves, Function minimization by conjugate gradients, Comput. J. 7 (1964)149–154.

[12] L. Grippo, S. Lucidi, A globally convergent version of the Polak–Ribi_ere conjugate gradient method, Math. Program. 78 (1997) 375–391.

[13] J.C. Gilbert, J. Nocedal, Global convergence properties of conjugate gradient methods for optimization, SIAM J. Optim. 2 (1) (1992) 21–42.

[14] M.R. Hestenes, Conjugate Direction Methods in Optimization, Springer-Verlag Inc., New York, 1980.

[15] D.M. Himmelblau, Applied Nonlinear Programming (Appendix A), McGraw-Hill Book Company, 1972.

[16] A. Miele, J.W. Cantrell, Studyon a memory gradient method for the minimization of functions, JOTA 3 (6) (1969) 459–470.

[17] J. Nocedal, J.W. Stephen, Numerical Optimization, Springer-Verlag Inc., New York, 1999.

[18] M.J.D. Powell, Restart procedures for the conjugate gradient method, Math. Program. 12 (1977) 241–254.

[19] M.J.D. Powell, Some convergence properties of the conjugate gradient method, Math. Program. 11 (1976) 42–49.

[20] Y.P. Qiu, Z.J. Shi, A new descent method for unconstrained optimization problem, Adv. Math. 29 (2) (2000) 183–184.

[21] Z.J. Shi, On super-memory gradient method with exact line searches, Asian Pacific J. Oper. Res. 20 (3) (2003).

[22] Z.J. Shi, A super memory gradient method for unconstrained optimization problem, J. Eng.Math. 17 (2) (2000) 99–104.

[23] J. Schropp, A note on minimization problems and multi step methods, Numer. Math. 78 (1997) 87–101.

[24] J.A. Syman, A new and dynamic method for unconstrained minimization, Appl. Math. Model.6 (1982) 449–462.

[25] M.N. Vrahatis, G.S. Androulakis, J.N. Lambrinos, G.D. Magoulas, A class of gradient unconstrained minimization algorithms with adaptive step size, J. Comput. Appl. Math. 114 (2000) 367–386.

[26] D.J. van Wyk, Differential optimization techniques, Appl. Math. Model. 8 (1984) 419–424.

[27] M.A. Wolfe, C. Viazminsky, Super memory descent methods for unconstrained minimization, JOTA 18 (4) (1976) 455–468.

[28] X. Wu, J. Xia, Z. Ouyang, Note on global convergence of ODE methods for unconstrained optimization, Appl. Math. Comput. 125 (2002) 311–315.