Journal of the Operations Research
Society of Japan
2005, Vol. 48, No. 4, 284-296

A NEW NONLINEAR CONJUGATE GRADIENT METHOD FOR UNCONSTRAINED OPTIMIZATION

Hiroshi Yabe
Tokyo University of Science

Naoki Sakaiwa
Hitachi Information Systems, Ltd.

(Received October 12, 2004; Revised May 10, 2005)

Abstract Conjugate gradient methods are widely used for large scale unconstrained optimization problems. Most of conjugate gradient methods don't always generate a descent search direction, so the descent condition is usually assumed in the analysis and implementations. Dai and Yuan (1999) proposed a conjugate gradient method which generates a descent search direction at every iteration and converges globally to the solution if the Wolfe conditions are satisfied within the line search strategy. In this paper, we give a new conjugate gradient method based on the study of Dai and Yuan, and show that our method always produces a descent search direction and converges globally if the Wolfe conditions are satisfied. Moreover our method has the second-order curvature information with a higher precision which uses the modified secant condition proposed by Zhang, Deng and Chen (1999) and Zhang and Xu (2001). Our numerical results show that our method is very efficient for given standard test problems, if we make a good choice of a parameter included in our method.

Keywords: Nonlinear programming, unconstrained optimization, conjugate gradient method, modified secant condition, global convergence

1. Introduction

We are concerned with the following unconstrained minimization problem:

$$\text{minimize } f(x), \tag{1.1}$$

where $f : \mathbf{R}^n \to \mathbf{R}$ is smooth and its gradient $g(x) \equiv \nabla f(x)$ is available. There are several kinds of numerical methods for solving (1.1), which include the steepest descent method, the Newton method and quasi-Newton methods, for example. Among them, the conjugate gradient method is one choice for solving large scale problems, because it does not need any matrices [4, 7]. Conjugate gradient methods are iterative methods and at the kth iteration, the form is given by

$$x_{k+1} = x_k + \alpha_k d_k, \tag{1.2}$$

where α_k is a positive step size and d_k is a search direction. Search directions are usually defined by

$$\begin{cases} d_0 &= -g_0 \\ d_{k+1} &= -g_{k+1} + \beta_{k+1} d_k \quad (k \geq 0), \end{cases} \tag{1.3}$$

where g_{k+1} denotes $\nabla f(x_{k+1})$ and $\beta_{k+1} \in \mathbf{R}$ is a scalar parameter, which characterizes conjugate gradient methods. Usually the parameter β_{k+1} is chosen so that (1.2)-(1.3) reduces to the linear conjugate gradient method if $f(x)$ is a strictly convex quadratic function and if α_k is calculated by the exact line search. Several kinds of formulas for β_{k+1} has been proposed. For example, the Fletcher-Reeves (FR), Polak-Ribiére-Polyak (PRP) and Hestenes-Stiefel (HS) formulas are well known and they are given by

$$\beta_{k+1}^{FR} = \frac{\|g_{k+1}\|^2}{\|g_k\|^2}, \quad (1.4)$$

$$\beta_{k+1}^{PRP} = \frac{g_{k+1}^T y_k}{\|g_k\|^2}, \quad (1.5)$$

$$\beta_{k+1}^{HS} = \frac{g_{k+1}^T y_k}{d_k^T y_k}, \quad (1.6)$$

where $\|\cdot\|$ denotes the Euclidean norm and

$$y_k = g_{k+1} - g_k.$$

The global convergence properties of the FR, PRP and HS methods without regular restarts have been studied by many researchers, including Al-Baali [1] and Gilbert and Nocedal [5]. The conjugate gradient method with regular restart was also found in [7]. To establish convergence properties of these methods, it is usually required that the step size α_k should satisfy the strong Wolfe conditions:

$$f(x_k) - f(x_k + \alpha_k d_k) \geq -\delta \alpha_k g_k^T d_k, \quad (1.7)$$

$$|g(x_k + \alpha_k d_k)^T d_k| \leq -\sigma g_k^T d_k, \quad (1.8)$$

where $0 < \delta < \sigma < 1$. On the other hand, many other numerical methods (e.g. the steepest descent method and quasi-Newton methods) for unconstrained optimization are proved to be convergent under the Wolfe conditions, which are weaker than the strong Wolfe conditions:

$$f(x_k) - f(x_k + \alpha_k d_k) \geq -\delta \alpha_k g_k^T d_k, \quad (1.9)$$

$$g(x_k + \alpha_k d_k)^T d_k \geq \sigma g_k^T d_k. \quad (1.10)$$

Thus it is an interesting issue to study global convergence of conjugate gradient methods under the Wolfe conditions instead of the strong Wolfe conditions.

Line search strategies require the descent condition

$$g_k^T d_k < 0 \quad \text{for all } k, \quad (1.11)$$

however most of conjugate gradient methods don't always generate a descent search direction, so condition (1.11) is usually assumed in the analysis and implementations. Some strategies have been studied which produce a descent search direction within the framework of conjugate gradient methods (see Fletcher [4] for example). Dai and Yuan [2] proposed the conjugate gradient method with

$$\beta_{k+1}^{DY} = \frac{\|g_{k+1}\|^2}{d_k^T y_k}, \quad (1.12)$$

which generates a descent search direction at every iteration and they showed that the proposed method converges globally if the Wolfe conditions (1.9) and (1.10) are satisfied. Furthermore, they dealt with a hybrid method in [3].

In this paper, we propose a new formula for β_{k+1}, which is motivated by the study of Dai and Yuan ([2], Section 2), and we show that our method always produces a descent search direction and converges globally if the Wolfe conditions (1.9) and (1.10) are satisfied. Moreover our formula has the second-order curvature information with a higher precision, which uses the modified secant condition proposed by Zhang, Deng and Chen [8] and Zhang and Xu [9]. Since our formula has available gradient, function value information and approximates Hessian information, we can expect our method to converge faster than the Dai-Yuan method (called the DY method).

This paper is organized as follows. In Section 2, we deal with an extension of the DY method and give a sufficient condition for generating a descent search direction. In Section 3, we establish global convergence of conjugate gradient methods discussed in Section 2. In Section 4, as one example of new methods, we give a conjugate gradient method which has second-order curvature information with a higher precision. In Section 5, some numerical experiments are presented. Our numerical results show that the proposed method is very efficient for given test problems, if we adopt a good choice of a parameter included in our formula.

2. Extension of the Dai-Yuan Method

In this section, we consider a condition that a descent search direction is generated, and we extend the DY method. We make such a direction inductively. Suppose that the current search direction d_k is a descent direction, namely, $g_k^T d_k < 0$ at the kth iteration. Now we need to find a β_{k+1} that produces a descent search direction d_{k+1}. This requires that

$$g_{k+1}^T d_{k+1} = -\|g_{k+1}\|^2 + \beta_{k+1} g_{k+1}^T d_k < 0. \tag{2.1}$$

Letting τ_{k+1} be a positive parameter, we define

$$\beta_{k+1} = \frac{\|g_{k+1}\|^2}{\tau_{k+1}}. \tag{2.2}$$

Equation (2.1) is equivalent to

$$\tau_{k+1} > g_{k+1}^T d_k.$$

Taking the positivity of τ_{k+1} into consideration, we have

$$\tau_{k+1} > \max\{g_{k+1}^T d_k, 0\}. \tag{2.3}$$

Therefore if condition (2.3) is satisfied for all k, the conjugate gradient method with (2.2) produces a descent search direction at every iteration. From (2.2), we can get various kinds of conjugate gradient methods by choosing various τ_{k+1}.

In Section 4, we will give a concrete τ_{k+1} satisfying (2.3) and establish global convergence of the proposed method. We note that the Wolfe condition (1.10) guarantees $d_k^T y_k > 0$ and that $d_k^T y_k = d_k^T g_{k+1} - d_k^T g_k > d_k^T g_{k+1}$. This implies that

$$d_k^T y_k > \max\{g_{k+1}^T d_k, 0\}. \tag{2.4}$$

By setting $\tau_{k+1} = d_k^T y_k$, formula (2.2) reduces to the DY method (1.12).

It follows from (1.3) and (2.2) that

$$\begin{aligned} g_{k+1}^T d_{k+1} &= -\|g_{k+1}\|^2 + \beta_{k+1} g_{k+1}^T d_k \\ &= -\tau_{k+1}\beta_{k+1} + \beta_{k+1} g_{k+1}^T d_k \\ &= (-\tau_{k+1} + g_{k+1}^T d_k)\beta_{k+1}. \end{aligned}$$

The above relation can be rewritten as

$$\beta_{k+1} = \frac{g_{k+1}^T d_{k+1}}{-\tau_{k+1} + g_{k+1}^T d_k}. \tag{2.5}$$

Recall that if we set $\tau_{k+1} = d_k^T y_k$, our method reduces to the DY method. Therefore formula (1.12) is also expressed as follows

$$\beta_{k+1}^{DY} = \frac{g_{k+1}^T d_{k+1}}{g_k^T d_k}. \tag{2.6}$$

Equations (2.5) and (2.6) play an important role in our convergence analysis in the next section.

3. Global Convergence of the Conjugate Gradient Method

In this section, we establish a convergence theorem of the conjugate gradient method with the parameter β_{k+1} given in (2.2). For this purpose, we make the following standard assumptions.

Assumption 3.1 (1) *f is bounded below on $\mathbf{R}^n$ and is continuously differentiable in a neighborhood $\mathcal{N}$ of the level set $\mathcal{L} = \{x \in \mathbf{R}^n \mid f(x) \le f(x_0)\}$ at the initial point x_0.*
(2) *The gradient $g(x)$ is Lipschitz continuous in $\mathcal{N}$, i.e., there exists a constant $L > 0$ such that*

$$\|g(x) - g(y)\| \le L\|x - y\|$$

for any $x, y \in \mathcal{N}$.

It should be noted that the assumption that the objective function is bounded below is weaker than the usual assumption that the level set is bounded. Under Assumption 3.1, we have the following useful lemma, which was proved in ([2], Lemma 3.2), for general iterative methods.

Lemma 3.2 *Suppose that Assumption* 3.1 *holds. Consider any method of the form* (1.2), *where d_k is a descent search direction and α_k satisfies the Wolfe conditions* (1.9) *and* (1.10). *Then the following holds*

$$\sum_{k=0}^{\infty} \frac{(g_k^T d_k)^2}{\|d_k\|^2} < \infty, \tag{3.1}$$

or equivalently

$$\sum_{k=0}^{\infty} \|g_k\|^2 \cos^2 \phi_k < \infty, \tag{3.2}$$

where ϕ_k is the angle between the search direction d_k and the steepest descent direction $-g_k$.

Note that condition (3.1) or (3.2) is known as the Zoutendijk condition.

By using this lemma, we obtain the following convergence theorem of the conjugate gradient method with (2.2).

Theorem 3.3 *Suppose that Assumption* 3.1 *holds. Let the sequence* $\{x_k\}$ *be generated by the conjugate gradient method with* β_{k+1} *in* (2.2).
(1) *If we choose* τ_{k+1} *such that* $\tau_{k+1} > \max\{g_{k+1}^T d_k,\ 0\}$ *for all k, then our method produces a descent search direction.*
(2) *If we choose* τ_{k+1} *such that* $\tau_{k+1} \geq d_k^T y_k$ *for all k, then our method either terminates at a stationary point or converges in the sense that*

$$\liminf_{k\to\infty} \|g_k\| = 0.$$

Proof. (1) The result follows directly from (2.3).
(2) By (2.4), we first note that $\tau_{k+1} \geq d_k^T y_k > \max\{g_{k+1}^T d_k,\ 0\}$. It is sufficient to consider only the case where the method does not terminate after finite many iterations. We will prove this theorem by contradiction. If the theorem is not true, there exists a constant $\bar{c} > 0$ such that

$$\|g_k\| \geq \bar{c} \quad \text{for all } k. \tag{3.3}$$

Rewriting (1.3) as

$$d_{k+1} + g_{k+1} = \beta_{k+1} d_k,$$

and squaring both sides of the above equation, we have

$$\|d_{k+1}\|^2 = \beta_{k+1}^2 \|d_k\|^2 - 2 g_{k+1}^T d_{k+1} - \|g_{k+1}\|^2.$$

Dividing both sides by $(g_{k+1}^T d_{k+1})^2$ and applying (2.5) yield

$$\begin{aligned}
\frac{\|d_{k+1}\|^2}{(g_{k+1}^T d_{k+1})^2} &= \frac{\beta_{k+1}^2 \|d_k\|^2}{(g_{k+1}^T d_{k+1})^2} - \frac{2}{g_{k+1}^T d_{k+1}} - \frac{\|g_{k+1}\|^2}{(g_{k+1}^T d_{k+1})^2} \\
&= \frac{\beta_{k+1}^2 \|d_k\|^2}{(g_{k+1}^T d_{k+1})^2} - \left(\frac{1}{\|g_{k+1}\|} + \frac{\|g_{k+1}\|}{g_{k+1}^T d_{k+1}}\right)^2 + \frac{1}{\|g_{k+1}\|^2} \\
&\leq \frac{\beta_{k+1}^2 \|d_k\|^2}{(g_{k+1}^T d_{k+1})^2} + \frac{1}{\|g_{k+1}\|^2} \\
&= \left(\frac{\beta_{k+1}^2 (g_k^T d_k)^2}{(g_{k+1}^T d_{k+1})^2}\right) \frac{\|d_k\|^2}{(g_k^T d_k)^2} + \frac{1}{\|g_{k+1}\|^2} \\
&= \left(\frac{(g_k^T d_k)^2}{(-\tau_{k+1} + g_{k+1}^T d_k)^2}\right) \frac{\|d_k\|^2}{(g_k^T d_k)^2} + \frac{1}{\|g_{k+1}\|^2}.
\end{aligned} \tag{3.4}$$

Noting that $\tau_{k+1} \geq d_k^T y_k$, we have

$$(\tau_{k+1} + g_k^T d_k - g_{k+1}^T d_k)(\tau_{k+1} - g_k^T d_k - g_{k+1}^T d_k) \ \geq \ 0.$$

By rearranging the above inequality, we have

$$\frac{(g_k^T d_k)^2}{(-\tau_{k+1} + g_{k+1}^T d_k)^2} \leq 1.$$

Hence, from (3.4), we get the following inequality

$$\frac{\|d_{k+1}\|^2}{(g_{k+1}^T d_{k+1})^2} \leq \frac{\|d_k\|^2}{(g_k^T d_k)^2} + \frac{1}{\|g_{k+1}\|^2}. \tag{3.5}$$

By noting $\|d_0\|^2/(g_0^T d_0)^2 = 1/\|g_0\|^2$, inequality (3.5) yields that

$$\frac{\|d_k\|^2}{(g_k^T d_k)^2} \le \sum_{i=0}^{k} \frac{1}{\|g_i\|^2} \quad \text{for all } k. \tag{3.6}$$

Therefore it follows from (3.6) and (3.3) that

$$\frac{(g_k^T d_k)^2}{\|d_k\|^2} \ge \frac{\bar{c}^2}{k+1},$$

which implies that

$$\sum_{k=0}^{\infty} \frac{(g_k^T d_k)^2}{\|d_k\|^2} = \infty.$$

This contradicts the Zoutendijk condition (3.1). Therefore the proof is complete. □

This theorem corresponds to Theorem 3.3 in [2] and Theorem 2.3 in [3].

Next we pay attention to the sufficient descent condition, namely, for some constant $c > 0$,

$$g_k^T d_k \le -c\|g_k\|^2 \quad \text{for all} \quad k \ge 0.$$

The sufficient descent condition is usually assumed in the analysis of conjugate gradient methods, while our method guarantees this condition if we choose τ_{k+1} such that $\tau_{k+1} \ge d_k^T y_k$ for all k and the strong Wolfe conditions (1.7)-(1.8) are satisfied at every iteration.

Theorem 3.4 *Suppose that Assumption* 3.1 *holds. Let the sequence* $\{x_k\}$ *be generated by our conjugate gradient method. If we choose* τ_{k+1} *such that*

$$\tau_{k+1} \ge d_k^T y_k \quad \text{for all} \quad k$$

and the step size α_k *satisfies the strong Wolfe conditions* (1.7)-(1.8)*, then our method satisfies the sufficient descent condition*

$$g_k^T d_k \le -c\|g_k\|^2 \quad \text{for all} \quad k, \tag{3.7}$$

with $c = \dfrac{1}{1+\sigma}$.

Proof. The proof is done by induction. By noting that $\sigma > 0$ implies $-1 < -\frac{1}{1+\sigma}$, the result clearly holds for $k = 0$ since

$$g_0^T d_0 = -\|g_0\|^2 \le -\frac{1}{1+\sigma}\|g_0\|^2 = -c\|g_0\|^2,$$

where $c = \frac{1}{1+\sigma}$.

Assume that (3.7) holds for some $k \ge 0$. It follows from the strong Wolfe condition (1.8) that

$$l_k \equiv \frac{g_{k+1}^T d_k}{g_k^T d_k} \in [-\sigma, \sigma] \quad \text{and} \quad d_k^T y_k > 0.$$

Then we have

$$l_k - 1 = \frac{g_{k+1}^T d_k}{g_k^T d_k} - 1 = \frac{g_{k+1}^T d_k - g_k^T d_k}{g_k^T d_k} = \frac{d_k^T y_k}{g_k^T d_k} < 0. \quad (3.8)$$

By (1.3), we have

$$g_{k+1}^T d_{k+1} = -\|g_{k+1}\|^2 + \beta_{k+1} g_{k+1}^T d_k.$$

Based on the sign of $g_{k+1}^T d_k$, we consider the following two cases.
(i) The case $g_{k+1}^T d_k \leq 0$: We immediately see that

$$\begin{aligned} g_{k+1}^T d_{k+1} &= -\|g_{k+1}\|^2 + \beta_{k+1} g_{k+1}^T d_k \\ &\leq -\|g_{k+1}\|^2 \\ &\leq -\frac{1}{1+\sigma}\|g_{k+1}\|^2. \end{aligned}$$

(ii) The case $g_{k+1}^T d_k > 0$: The conditions on β_{k+1} and (3.8) yield

$$\begin{aligned} g_{k+1}^T d_{k+1} &= -\|g_{k+1}\|^2 + \beta_{k+1} g_{k+1}^T d_k \\ &\leq -\|g_{k+1}\|^2 + \frac{\|g_{k+1}\|^2}{d_k^T y_k} g_{k+1}^T d_k \\ &= \left(-1 + \frac{g_{k+1}^T d_k}{d_k^T y_k}\right)\|g_{k+1}\|^2 \\ &= \left(\frac{g_k^T d_k}{d_k^T y_k}\right)\|g_{k+1}\|^2 \\ &= \frac{1}{l_k - 1}\|g_{k+1}\|^2. \end{aligned}$$

By noting $\frac{1}{1+\sigma} \leq \frac{1}{1-l_k} \leq \frac{1}{1-\sigma}$, the inequality above implies that

$$\begin{aligned} g_{k+1}^T d_{k+1} &\leq -\frac{1}{1-l_k}\|g_{k+1}\|^2 \\ &\leq -\frac{1}{1+\sigma}\|g_{k+1}\|^2. \end{aligned}$$

By summarizing the cases (i) and (ii), the sufficient descent condition (3.7) holds with $c = \frac{1}{1+\sigma}$ at the $k+1$st iteration. Therefore the theorem is proved. □

4. New Conjugate Gradient Method Based on the Modified Secant Condition

In this section, we present a concrete formula of τ_{k+1} that satisfies condition (2.3) and propose a new conjugate gradient method. We also show the global convergence of the proposed method by using Theorem 3.3.

In order to accelerate the conjugate gradient method, we aim to incorporate the second order information of the objective function into the formula of β_{k+1}. For this purpose, we make use of secant conditions of quasi-Newton methods. The quasi-Newton method is known as another method for solving unconstrained optimization problem (1.1). This method generates sequences $\{x_k\}$ and $\{B_k\}$ by the iteration $x_{k+1} = x_k + \alpha_k d_k$ and an updating formula for B_k, where d_k is a search direction that is obtained by solving the linear system of equations $B_k d_k = -g_k$ and B_k is an approximation to the Hessian $\nabla^2 f(x_k)$ of $f(x)$ at x_k. A new matrix B_{k+1} is usually required to satisfy the secant condition

$$B_{k+1}s_k = y_k, \tag{4.1}$$

where

$$s_k = x_{k+1} - x_k \quad \text{and} \quad y_k = g_{k+1} - g_k.$$

The above equation employs only the gradients, but doesn't have the available function value information. By extending the secant condition (4.1), Zhang, Deng and Chen [8] and Zhang and Xu [9] proposed the modified secant condition

$$\widehat{B}_{k+1}s_k \;=\; \widehat{y}_k \tag{4.2}$$

with

$$\begin{cases} \widehat{y}_k &= y_k + \dfrac{\theta_k}{s_k^T u_k} u_k \\ \theta_k &= 6(f(x_k) - f(x_{k+1})) + 3(g_k + g_{k+1})^T s_k, \end{cases} \tag{4.3}$$

where u_k is any vector with $s_k^T u_k \neq 0$ and $\widehat{B}_{k+1}$ is a reasonable approximation to the Hessian $\nabla^2 f(x_{k+1})$. It should be noted that the modified secant condition (4.2) has both available gradient and function value information. The following result which was given in [8, 9] plays an important role when we apply the modified secant condition to our conjugate gradient method. Though the proof can be found in [9], we give a brief proof for completeness.

Lemma 4.1 *Assume that the function $f(x)$ is sufficiently smooth. If $\|s_k\|$ is sufficiently small, then for any vector u with $s_k^T u_k \neq 0$, we have*

$$s_k^T[\nabla^2 f(x_{k+1})s_k - \widehat{y}_k] \;=\; O(\|s_k\|^4), \tag{4.4}$$
$$s_k^T[\nabla^2 f(x_{k+1})s_k - y_k] \;=\; O(\|s_k\|^3). \tag{4.5}$$

Proof. It follows from the Taylor expansion that

$$\begin{aligned} f(x_k) &= f(x_{k+1}) - g_{k+1}^T s_k + \frac{1}{2} s_k^T \nabla^2 f(x_{k+1}) s_k - \frac{1}{6} s_k^T (T_{k+1} s_k) s_k + O(\|s_k\|^4), \\ s_k^T g_k &= s_k^T g_{k+1} - s_k^T \nabla^2 f(x_{k+1}) s_k + \frac{1}{2} s_k^T (T_{k+1} s_k) s_k + O(\|s_k\|^4), \end{aligned}$$

where $T_{k+1} \in \mathbf{R}^{n\times n\times n}$ is the tensor of f at x_{k+1}. By canceling the terms which include the tensor, we have

$$\begin{aligned} s_k^T \nabla^2 f(x_{k+1}) s_k &= (g_{k+1} - g_k)^T s_k + 6(f(x_k) - f(x_{k+1})) + 3(g_k + g_{k+1})^T s_k + O(\|s_k\|^4) \\ &= s_k^T y_k + \theta_k + O(\|s_k\|^4). \end{aligned}$$

Since the following hold

$$f(x_{k+1}) = f(x_k) + g_k^T s_k + \frac{1}{2} s_k^T \nabla^2 f(x_k) s_k + O(\|s_k\|^3)$$

and

$$g_{k+1}^T s_k = g_k^T s_k + s_k^T \nabla^2 f(x_k) s_k + O(\|s_k\|^3),$$

we see that

$$\begin{aligned}\theta_k &= 6\left(-g_k^T s_k - \frac{1}{2}s_k^T\nabla^2 f(x_k)s_k + O(\|s_k\|^3)\right) + 3g_k^T s_k + 3(g_k^T s_k + s_k^T\nabla^2 f(x_k)s_k + O(\|s_k\|^3)) \\ &= O(\|s_k\|^3).\end{aligned}$$

Therefore we obtain

$$s_k^T\nabla^2 f(x_{k+1})s_k - s_k^T y_k = \theta_k + O(\|s_k\|^4) = O(\|s_k\|^3)$$

and

$$s_k^T\nabla^2 f(x_{k+1})s_k - s_k^T \hat{y}_k = O(\|s_k\|^4).$$

The proof is complete. □

It follows from (4.4) and (4.5) that

$$\begin{aligned}s_k^T\widehat{y}_k &= s_k^T\nabla^2 f(x_{k+1})s_k + O(\|s_k\|^4), \\ s_k^T y_k &= s_k^T\nabla^2 f(x_{k+1})s_k + O(\|s_k\|^3).\end{aligned}$$

These two estimates show that $s_k^T\widehat{y}_k$ approximates the second-order curvature $s_k^T\nabla^2 f(x_{k+1})s_k$ with a higher precision than $s_k^T y_k$ dose.

By premultiplying (4.3) by d_k^T, we have

$$d_k^T\widehat{y}_k = d_k^T y_k + \frac{\theta_k}{s_k^T u_k}d_k^T u_k, \tag{4.6}$$

and we would like to apply this quantity to the parameter τ_{k+1} in our conjugate gradient method (2.2). However, since the second term in (4.6) is not always non-negative, Theorem 3.3 does not always hold, when we directly choose $\tau_{k+1} = d_k^T\widehat{y}_k$. In order to remedy this difficulty, we set

$$\tau_{k+1}^{new} = d_k^T y_k + t_k\max\left\{\frac{\theta_k}{s_k^T u_k}d_k^T u_k, 0\right\},$$

where $t_k \geq 0$. By noting that $s_k = \alpha_k d_k$, the above equation is rewritten as

$$\tau_{k+1}^{new} = d_k^T y_k + \frac{t_k}{\alpha_k}\max\{\theta_k, 0\}. \tag{4.7}$$

Note that (4.7) does not depend on a choice of u_k. One reasonable value of t_k in (4.7) is $t_k = 1$, because this case corresponds to (4.6). If we set $t_k = 0$, (4.7) reduces to $\tau_{k+1}^{new} = d_k^T y_k$, namely the Dai and Yuan formula (1.12). Following the proof of Lemma 4.1, we have

$$s_k^T\nabla^2 f(x_{k+1})s_k = s_k^T y_k + \theta_k + O(\|s_k\|^4) \quad \text{and} \quad |\theta_k| = O(\|s_k\|^3).$$

Thus we obtain

$$\begin{aligned}&|\alpha_k\tau_{k+1}^{new} - s_k^T\nabla^2 f(x_{k+1})s_k| \\ &= |t_k\max\{\theta_k, 0\} - \theta_k + O(\|s_k\|^4)| \\ &\leq \begin{cases} |(t_k-1)\theta_k| + O(\|s_k\|^4) \leq |t_k - 1|O(\|s_k\|^3) + O(\|s_k\|^4) & (\text{if } \theta_k > 0), \\ |\theta_k| + O(\|s_k\|^4) = O(\|s_k\|^3) & (\text{otherwise}).\end{cases}\end{aligned} \tag{4.8}$$

This implies that $\alpha_k\tau_{k+1}^{new}$ has a good information of the second order curvature $s_k^T\nabla^2 f(x_{k+1})s_k$.

We now state the algorithm of our conjugate gradient method as follows.

Algorithm 4.1

Step0. Given $x_0 \in \mathbf{R}^n$, set $d_0 = -g_0$, $k := 0$, and if $g_0 = 0$, then stop.
Step1. Compute the step size $\alpha_k > 0$ satisfying (1.9) and (1.10).
Step2. Let $x_{k+1} = x_k + \alpha_k d_k$. If $g_{k+1} = 0$, then stop.
Step3. Calculate t_k and θ_k in (4.3). Compute τ_{k+1}^{new} by (4.7) and set

$$\beta_{k+1} = \frac{\|g_{k+1}\|^2}{\tau_{k+1}^{new}}. \tag{4.9}$$

Step4. Generate d_{k+1} by $d_{k+1} = -g_{k+1} + \beta_{k+1} d_k$.
Step5. Set $k := k + 1$ and go to Step1.

Since $\tau_{k+1}^{new} \geq d_k^T y_k$ is satisfied for all k, the assumptions of Theorem 3.3 always hold. Therefore, we obtain the following theorem that implies the global convergence of our method.

Theorem 4.2 *Suppose that Assumption* 3.1 *holds. Let the sequence* $\{x_k\}$ *be generated by Algorithm 4.1. Then our method always produces a descent search direction, and either terminates at a stationary point or converges in the sense that*

$$\liminf_{k \to \infty} \|g_k\| = 0.$$

From Lemma 4.1 and the discussion above, τ_{k+1}^{new} can be expected to approximate $\alpha_k (d_k^T \nabla^2 f(x_{k+1}) d_k)$ with a higher precision than $d_k^T y_k$ does. Moreover τ_{k+1}^{new} has both available gradient and function value information. Therefore, we can expect that our method performs better than the DY method (1.12).

Closing this section, we note that our method reduces to the Hestenes-Stiefel method within the framework of the linear conjugate gradient method, because the facts $\theta_k = 0$ and $g_{k+1}^T g_k = 0$ yield $\beta_{k+1} = \dfrac{g_{k+1}^T g_{k+1}}{d_k^T y_k} = \dfrac{g_{k+1}^T y_k}{d_k^T y_k}$ for a strictly convex quadratic function.

5. Preliminary Numerical Results

We compare our method (4.9) with the FR method (1.4), PRP method (1.5), HS method (1.6) and the DY method (1.12) from the viewpoint of numerical performance. The test problems are drawn from the paper [6]. The preliminary numerical results of our tests are reported in Table 5.1. The first column "P" and the second column "Name" denote the problem number and problem name, respectively. Each problem was tested with different values of n ranging from $n = 20$ to $n = 10000$. We give only two cases of n in Table 5.1. The numerical results are given in the form of I/F, where I and F denote the numbers of iterations and function value evaluations respectively. The stopping condition is

$$\|g_k\|_\infty \leq 10^{-5}. \tag{5.1}$$

We also terminate the iteration if function value improvement is too small. More exactly, iterations are terminated whenever

$$0 \leq f(x_k) - f(x_{k+1}) \leq 10^{-10}(1 + |f(x_k)|). \tag{5.2}$$

In this case, we use a superscript "*" to show that the iteration is terminated due to (5.2) but (5.1) is not satisfied. In addition, we write "Failed" if d_k is so large that a numerical overflow occurs while the method tries to compute $f(x_k + d_k)$. If the number of iterations exceeds 1000, we also stop the algorithm. However this case didn't occur in our numerical experiments. Our line search subroutine computes α_k such that the Armijo rule (1.9) holds with $\delta = 0.01$. The initial value of α_k is always set to 1 in the line search procedure at each iteration. Although our line search cannot always ensure a descent search direction for all methods, an uphill search direction did not occur in our numerical experiments. For method (4.9), we fixed t_k to t independent of k, and tested values of t ranging from 0 to 100, and in Table 5.1 "θ" denotes the number of iterations for which $\theta_k > 0$ holds. In this case, (4.6) is employed.

From Table 5.1, we see that for some problems, method (4.9) with $t = 1$ really performs much better than the DY method, for example Problem 21 with $n = 1000$ and $n = 10000$. However, for some other problems, method (4.9) with $t = 1$ performs worse than the DY method, for example Problem 22 with $n = 1000$ and $n = 10000$. On the whole, method (4.9) with $t = 1$ and the DY method perform quite similarly for the given test problems. In case of Problem 25, the case $\theta = 0$ means that results of method (4.9) coincide with those of the DY method. We found a good choice of a parameter t_k by testing values of t ranging from 0 to 100. Good choices are listed in the last column of Table 5.1. Specifically, in the case $t = 17$ for Problem 21, the numbers of iterations and function evaluations are 15 and 48 respectively, which was the best case of our numerical tests for this problem. Furthermore, method (4.9) with $t = 99.6$ was the best case for Problem 22. However, we did not observe a special tendency of a choice of t_k.

If we make a good choice of parameter t_k, method (4.9) outperforms the other methods. We also note that numerical performance depends on how often positive values of θ_k are adopted.

6. Conclusion

In this paper, we have considered the conjugate gradient method with the formula (2.2) and have proved that it converges globally to the solution if we choose τ_{k+1} such that $\tau_{k+1} \geq d_k^T y_k$ for all k and the Wolfe conditions are satisfied. This method includes the Dai-Yuan method as a special case. We have also shown that the search direction satisfied the sufficient descent condition if we used the strong Wolfe conditions. As a choice of concrete τ_{k+1}, we have proposed (4.7) based on the modified secant condition (4.2). The conjugate gradient method with (4.7) always produces descent search directions and converges globally. It should be noted that our formula has the available gradient, function value information, and the second order curvature information.

The preliminary numerical experiments show that if we choose a good value of t_k, method (4.9) performs very well. However we have not theoretically estimated an optimal parameter t_k yet. In our preliminary numerical experiments, we fixed the value of t_k. It is interesting to make a suitable choice of the parameter t_k based on the estimate (4.8). We consider a parameter such that t_k approaches 1, for example. Theoretical or practical choice for t_k is further research.

Table 5.1 : Numerical comparisons

P	Name	n	FR method I/F	PRP Method I/F	HS Method I/F
21	Extended Rosenbrock	1000	64/219	33/154	25/211
		10000	62/214	33/154	25/211
22	Extended Powell	1000	179/512	290/847	*Failed*
		10000	222/620	242/661	252/615
26	Trigonometric	100	3/4	7/8	7/8
		1000	2/3	5/6	6/8
23	Penalty I	100	*Failed*	*Failed*	*Failed*
		1000	50/95	19/150	38/189
24	Penalty II	20	77/208*	107/246*	76/170*
		50	125/455*	92/276*	175/444*
30	Broyden Tridiagonal	100	41/123	40/120	41/123
		1000	95/283	87/296	49/146
25	Variably dimensioned	100	7/71	17/221	16/191
		1000	14/203	18/289	31/234

P	Name	n	DY method I/F	Method (4.9) with t = 1. I/F/θ	Method (4.9) with good t. I/F/θ
21	Extended Rosenbrock	1000	53/169	35/109/17	15/48/8(t = 17)
		10000	53/169	35/109/17	15/48/8(t = 17)
22	Extended Powell	1000	125/341	209/610/82	26/80/11(t = 99.6)
		10000	147/424	181/522/66	26/80/11(t = 99.6)
26	Trigonometric	100	3/4	3/4/2	3/4/2(t = 1)
		1000	3/4	3/4/2	3/4/2(t = 1)
23	Penalty I	100	40/91	37/70/17	37/70/17(t = 1)
		1000	26/215	28/124/15	22/121/8(t = 100)
24	Penalty II	20	62/161*	79/179/31*	24/70/13(t = 17)*
		50	114/336*	126/37/71*	92/273/49(t = 19)*
30	Broyden Tridiagonal	100	36/111	36/108/16	35/108/16(t = 3)
		1000	59/178	60/181/36	50/151/44(t = 5)
25	Variably dimensioned	100	9/88	9/88/0	9/88/0(t = 1)
		1000	16/234	16/234/0	16/234/0(t = 1)

Acknowledgements

The authors would like to thank two anonymous referees for their comments to improve the paper. The first author is supported in part by the Grant-in-Aid for Scientific Research (C) 16510123 of the Ministry of Education, Culture, Sports, Science and Technology of Japan.

References

[1] M. Al-Baali: Descent property and global convergence of the Fletcher-Reeves method with inexact line search. *IMA Journal of Numerical Analysis*, **5** (1985), 121-124.

[2] Y.H. Dai and Y. Yuan: A nonlinear conjugate gradient method with a strong global convergence property. *SIAM Journal on Optimization*, **10** (1999), 177-182.

[3] Y.H. Dai and Y. Yuan: An efficient hybrid conjugate gradient method for unconstrained optimization. *Annals of Operations Research*, **103** (2001), 33-47.

[4] R. Fletcher: *Practical Methods of Optimization, 2nd ed.* (John Wiley & Sons, New York, 1987).

[5] J.C. Gilbert and J. Nocedal: Global convergence properties of conjugate gradient methods for optimization. *SIAM Journal on Optimization*, **2** (1992), 21-42.

[6] J.J. Moré, B.S. Garbow and K.E. Hillstrom: Testing unconstrained optimization software. *ACM Transactions on Mathematical Software*, **7** (1981), 17-41.

[7] J. Nocedal and S.J. Wright: *Numerical Optimization* (Springer Verlag, New York, 1999).

[8] J.Z. Zhang, N.Y. Deng and L.H. Chen: New quasi-Newton equation and related methods for unconstrained optimization. *Journal of Optimization Theory and Applications*, **102** (1999), 147-167.

[9] J. Zhang and C. Xu: Properties and numerical performance of quasi-Newton methods with modified quasi-Newton equations. *Journal of Computational and Applied Mathematics*, **137** (2001), 269-278.

Hiroshi Yabe
Department of Mathematical Information Science
Tokyo University of Science
1-3 Kagurazaka, Shinjuku-ku
Tokyo 162-8601, Japan
E-mail: `yabe@rs.kagu.tus.ac.jp`