Research Article

A Descent Dai-Liao Conjugate Gradient Method Based on a Modified Secant Equation and Its Global Convergence

Ioannis E. Livieris and Panagiotis Pintelas

Department of Mathematics, University of Patras, 265-00 Patras, Greece

Correspondence should be addressed to Ioannis E. Livieris, livieris@upatras.gr

Received 31 August 2011; Accepted 20 October 2011

Academic Editors: K. Eom and R. Joan-Arinyo

Copyright © 2012 I. E. Livieris and P. Pintelas. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

We propose a conjugate gradient method which is based on the study of the Dai-Liao conjugate gradient method. An important property of our proposed method is that it ensures sufficient descent independent of the accuracy of the line search. Moreover, it achieves a high-order accuracy in approximating the second-order curvature information of the objective function by utilizing the modified secant condition proposed by Babaie-Kafaki et al. (2010). Under mild conditions, we establish that the proposed method is globally convergent for general functions provided that the line search satisfies the Wolfe conditions. Numerical experiments are also presented.

1. Introduction

We consider the unconstrained optimization problem:

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

where $f : \mathbb{R}^n \rightarrow \mathbb{R}$ is a continuously differentiable function. Conjugate gradient methods are probably the most famous iterative methods for solving the optimization problem (1), especially when the dimension is large characterized by the simplicity of their iteration and their low memory requirements. These methods generate a sequence of points $\{x_k\}$, starting from an initial point $x_0 \in \mathbb{R}^n$, using the iterative formula:

$$x_{k+1} = x_k + \alpha_k d_k, \quad k = 0, 1, \dots, \quad (2)$$

where $\alpha_k > 0$ is the stepsize obtained by some line search and d_k is the search direction defined by

$$d_k = \begin{cases} -g_0, & \text{if } k = 0; \\ -g_k + \beta_k d_{k-1}, & \text{otherwise,} \end{cases} \quad (3)$$

where g_k is the gradient of f at x_k and β_k is a scalar. Well-known formulas for β_k include the Hestenes-Stiefel (HS) [1],

the Fletcher-Reeves (FR) [2], the Polak-Ribière (PR) [3], the Liu-Storey (LS) [4], the Dai-Yuan (DY) [5] and the conjugate descent (CD) [6]. They are specified by

$$\begin{aligned} \beta_k^{\text{HS}} &= \frac{g_k^T y_{k-1}}{y_{k-1}^T d_{k-1}}, & \beta_k^{\text{FR}} &= \frac{\|g_k\|^2}{\|g_{k-1}\|^2}, \\ \beta_k^{\text{PR}} &= \frac{g_k^T y_{k-1}}{\|g_{k-1}\|^2}, & \beta_k^{\text{LS}} &= -\frac{g_k^T y_{k-1}}{d_{k-1}^T g_{k-1}}, \\ \beta_k^{\text{DY}} &= \frac{\|g_k\|^2}{y_{k-1}^T d_{k-1}}, & \beta_k^{\text{CD}} &= -\frac{\|g_k\|^2}{d_{k-1}^T g_{k-1}}, \end{aligned} \quad (4)$$

respectively, where $s_{k-1} = x_k - x_{k-1}$, $y_{k-1} = g_k - g_{k-1}$, and $\|\cdot\|$ denotes the Euclidean norm. If f is a strictly convex quadratic function, and the performed line search is exact, all these methods are equivalent, but for a general function different choices of β_k give rise to distinct conjugate gradient methods with quite different computational efficiency and convergence properties. We refer to the books [7, 8], the survey paper [9], and the references therein about the numerical performance and the convergence properties of conjugate gradient methods. During the last decade, much effort has been devoted to develop new conjugate gradient methods which are not only globally convergent for general

functions but also computationally superior to classical methods and there are classified by two classes.

The first class utilizes the second-order information of the objective function to improve the efficiency and robustness of conjugate gradient methods. Dai and Liao [10] proposed a new conjugate gradient method by exploiting a new conjugacy condition based on the standard secant equation in which β_k in (3) is defined by

$$\beta_k^{\text{DL}} = \frac{g_k^T (y_{k-1} - t s_{k-1})}{d_{k-1}^T y_{k-1}}, \quad (5)$$

where $t \geq 0$ is a scalar. Moreover, Dai and Liao also suggested a modification of (5) from a viewpoint of global convergence for general functions, by restricting the first term of being nonnegative, namely,

$$\beta_k^{\text{DL+}} = \max \left\{ \frac{g_k^T y_{k-1}}{d_{k-1}^T y_{k-1}}, 0 \right\} - t \frac{g_k^T s_{k-1}}{d_{k-1}^T y_{k-1}}. \quad (6)$$

Along this line, many researchers [11–15] proposed variants of Dai-Liao method based on modified secant conditions with higher orders of accuracy in the approximation of the curvature. Under proper conditions, these methods are globally convergent and sometimes competitive to classical conjugate gradient methods. However, these methods do not ensure to generate descent directions, therefore, the descent condition is usually assumed in their analysis and implementations.

The second class focuses to generate conjugate gradient methods which ensure sufficient descent independent of the accuracy of the line search. On the basis of this idea, Hager and Zhang [16] considered to modify parameter β_k in (3) and proposed a new conjugate gradient method, called CG-DESCENT in which the update parameter is defined as follows:

$$\beta_k^{\text{HZ}} = \max \{ \beta_k^{\text{N}}, \eta_k \}, \quad (7)$$

where

$$\beta_k^{\text{N}} = \frac{g_k^T y_{k-1}}{d_{k-1}^T y_{k-1}} - 2 \frac{\|y_{k-1}\|^2 g_k^T d_{k-1}}{(d_{k-1}^T y_{k-1})^2}, \quad (8)$$

$$\eta_k = - \frac{1}{\|d_k\|^2 \min \{ \|g_k\|, \eta \}},$$

and $\eta > 0$ is a constant. An important feature of the CG-DESCENT method is that the generated direction satisfies $d_k^T g_k \leq -(7/8)\|g_k\|^2$. Moreover, Hager and Zhang [16] established that the CG-DESCENT is globally convergent for general functions under the Wolfe line search conditions.

Quite recently, Zhang et al. [17] considered a different approach, to modify the search direction such that the generated direction satisfies $g_k^T d_k = -\|g_k\|^2$, independently of the line search used. More analytically, they proposed a modified FR method in which the search direction is given by

$$d_k = - \left(1 + \beta_k^{\text{FR}} \frac{g_k^T d_{k-1}}{\|g_k\|} \right) g_k + \beta_k^{\text{FR}} d_{k-1}. \quad (9)$$

This method is reduced to the standard FR method in case the performed line search is exact. Furthermore, in case β_k is specified by another existing conjugate gradient formula, the property $g_k^T d_k = -\|g_k\|^2$ is still satisfied. Along this line, many related conjugate gradient methods have been extensively studied [17–24] with strong convergence properties and good average performance.

In this work, we propose a new conjugate gradient method which has both characteristics of the two previously discussed classes. More analytically, our method ensures sufficient descent independent of the accuracy of the line search and achieves a high-order accuracy in approximating the second-order curvature information of the objective function by utilizing the modified secant condition proposed by Babaie-Kafaki et al. [11]. Under mild conditions, we establish the global convergence of our proposed method. The numerical experiments indicate that the proposed method is promising.

The remainder of this paper is organized as follows. In Section 2, we present our motivation and our proposed conjugate gradient method. In Section 3, we present the global convergence analysis of our method. The numerical experiments are reported in Section 4 using the performance profiles of Dolan and Moré [25]. Finally, Section 4 presents our concluding remarks and our proposals for future research.

Throughout this paper, we denote $f(x_k)$ and $\nabla^2 f(x_k)$ as f_k and G_k , respectively.

2. Algorithm

Firstly, we recall that for quasi-Newton methods, an approximation matrix B_{k-1} to the Hessian $\nabla^2 f_{k-1}$ is updated so that a new matrix B_k satisfies the following secant condition:

$$B_k s_{k-1} = y_{k-1}. \quad (10)$$

Zhang et al. [26] and Zhang and Xu [27] expanded this condition and derived a class of modified secant condition with a vector parameter, in the form

$$B_k s_{k-1} = \tilde{y}_{k-1}, \quad \tilde{y}_{k-1} = y_{k-1} + \frac{\theta_{k-1}}{s_{k-1}^T u} u, \quad (11)$$

where u is any vector satisfying $s_{k-1}^T u > 0$, and θ_{k-1} is defined by

$$\theta_{k-1} = 6(f_{k-1} - f_k) + 3(g_k + g_{k-1})^T s_{k-1}. \quad (12)$$

Observing that this new quasi-Newton equation contains not only gradient value information but also function value information at the present and the previous step. Moreover, in [26], Zhang et al. proved that if $\|s_{k-1}\|$ is sufficiently small, then

$$s_{k-1}^T (G_k s_{k-1} - y_{k-1}) = O(\|s_{k-1}\|^3),$$

$$s_{k-1}^T (G_k s_{k-1} - \tilde{y}_{k-1}) = O(\|s_{k-1}\|^4). \quad (13)$$

Clearly, these equations imply that the quantity $s_{k-1}^T \tilde{y}_{k-1}$ approximates the second order curvature $s_{k-1}^T G_k s_{k-1}$ with a higher precision than the quantity $s_{k-1}^T y_{k-1}$ does.

However, for values of $\|s_{k-1}\|$ greater than one (i.e., $\|s_{k-1}\| > 1$), the standard secant equation (10) is expected to be more accurate than the modified secant equation (11). Recently, Babaie-Kafaki et al. [11] to overcome this difficulty considered an extension of the modified secant (11) as follows:

$$\begin{aligned} B_k s_{k-1} &= \tilde{y}_{k-1}^*, \\ \tilde{y}_{k-1}^* &= y_{k-1} + \rho_{k-1} \frac{\max\{\theta_{k-1}, 0\}}{s_{k-1}^T u} u, \end{aligned} \quad (14)$$

where parameter ρ_{k-1} is restricted to values of $\{0, 1\}$ and adaptively switch between the standard secant equation (10) and the modified secant equation (14), by setting $\rho_{k-1} = 1$ if $\|s_{k-1}\| \leq 1$ and setting $\rho_{k-1} = 0$, otherwise. In the same way as Dai and Liao [10], they obtained an expression for β_k , in the form

$$\tilde{\beta}_k = \frac{g_k^T (\tilde{y}_{k-1}^* - t s_{k-1})}{d_{k-1}^T \tilde{y}_{k-1}^*}, \quad (15)$$

where $t \geq 0$ and $\tilde{y}_{k-1}^*$ is defined by (14). Furthermore, following Dai and Liao's approach, in order to ensure global convergence for general functions, they modified formula (15) as follows:

$$\tilde{\beta}_k^* = \max \left\{ \frac{g_k^T \tilde{y}_{k-1}^*}{d_{k-1}^T \tilde{y}_{k-1}^*}, 0 \right\} - t \frac{g_k^T s_{k-1}}{d_{k-1}^T \tilde{y}_{k-1}^*}. \quad (16)$$

Motivated by the theoretical advantages of the modified secant equation (14) and the technique of the modified FR method [17], we propose a new conjugate gradient method as follows. Let the search direction be defined by

$$d_k = - \left(1 + \beta_k \frac{g_k^T d_{k-1}}{\|g_k\|} \right) g_k + \beta_k d_{k-1}, \quad (17)$$

where β_k is defined by (15). It is easy to see that the sufficient descent condition holds:

$$g_k^T d_k = -\|g_k\|^2, \quad (18)$$

using any line search. Moreover, if f is a convex quadratic function and the performed line search is exact, then $\theta_{k-1} = 0$, $\tilde{y}_{k-1}^* = y_{k-1}$, and $g_k^T s_{k-1} = 0$; hence, the conjugate gradient method (2)–(17) is reduced to the standard conjugate gradient method, accordingly.

Now, based on the above discussion, we present our proposed algorithm called modified Dai Liao conjugate gradient algorithm (MDL-CG).

3. Convergence Analysis

In order to establish the global convergence analysis, we make the following assumptions for the objective function f .

1. Choose an initial point $x_0 \in \mathbb{R}^n$; Set $k = 0$.
2. If $\|g_k\| = 0$, then terminate; Otherwise go to the next.
3. Compute the descent direction d_k by (15) and (17).
4. Determine a stepsize α_k by some line search rule.
5. Let $x_{k+1} = x_k + \alpha_k d_k$.
6. Set $k = k + 1$ and go to 2.

ALGORITHM 1: (MDL-CG).

Assumption 1. The level set $\mathcal{L} = \{x \in \mathbb{R}^n \mid f(x) \leq f(x_0)\}$ is bounded, namely, there exists a positive constant $B > 0$ such that

$$\|x\| \leq B, \quad \forall x \in \mathcal{L}. \quad (19)$$

Assumption 2. In some neighborhood $\mathcal{N} \in \mathcal{L}$, f is differentiable and its gradient g is Lipschitz continuous, namely, there exists a positive constant $L > 0$ such that

$$\|g(x) - g(y)\| \leq L\|x - y\|, \quad \forall x, y \in \mathcal{N}. \quad (20)$$

It follows directly from Assumptions 1 and 2 that there exists a positive constant $M > 0$ such that

$$\|g(x)\| \leq M, \quad \forall x \in \mathcal{L}. \quad (21)$$

In order to guarantee the global convergence of Algorithm 1, we will impose that the steplength α_k satisfies the Armijo conditions or the Wolfe conditions. The Armijo line search is to find a steplength $\alpha_k = \max\{\lambda^j, j = 0, 1, 2, \dots\}$ such that

$$f(x_k + \alpha_k d_k) - f(x_k) \leq \delta \alpha_k g_k^T d_k, \quad (22)$$

where $\delta, \lambda \in (0, 1)$ are constants. In the Wolfe line search the steplength α_k satisfies

$$f(x_k + \alpha_k d_k) - f(x_k) \leq \sigma_1 \alpha_k g_k^T d_k, \quad (23)$$

$$g(x_k + \alpha_k d_k)^T d_k \geq \sigma_2 g_k^T d_k, \quad (24)$$

with $0 < \sigma_1 < \sigma_2 < 1$.

Next, we present some lemmas which are very important for the global convergence analysis.

Lemma 1 (see [11]). *Suppose that Assumptions 1 and 2 hold. For θ_{k-1} and $\tilde{y}_{k-1}^*$ defined by (12) and (14), respectively, one has*

$$|\theta_{k-1}| \leq 3L\|s_{k-1}\|^2, \quad \|\tilde{y}_{k-1}^*\| \leq 4L\|s_{k-1}\|. \quad (25)$$

Lemma 2. *Suppose that Assumptions 1 and 2 hold. Let $\{x_k\}$ be generated by Algorithm MDL-CG, where the line search satisfies the Armijo condition (22), then there exists a positive constant $c > 0$ such that*

$$\alpha_k \geq c \frac{\|g_k\|^2}{\|d_k\|^2}, \quad (26)$$

for all $k \geq 0$.

Proof. From the Armijo condition (22) and Assumptions 1 and 2, we have

$$\sum_{k \geq 0} -\delta \alpha_k d_k^T g_k < +\infty. \quad (27)$$

Using this together with inequality (18) implies that

$$\sum_{k \geq 0} \alpha_k \|g_k\|^2 = -\sum_{k \geq 0} \alpha_k d_k^T g_k < +\infty. \quad (28)$$

We now prove (26) by considering the following cases.

Case 1. If $\alpha_k = 1$, it follows from (18) that $\|d_k\| \leq \|g_k\|$. In this case, inequality (26) is satisfied with $c = 1$.

Case 2. If $\alpha_k < 1$ by the line search, $\lambda^{-1}\alpha_k$ does not satisfy (22). This implies

$$f(x_k + \lambda^{-1}\alpha_k d_k) - f(x_k) > \delta \lambda^{-1}\alpha_k g_k^T d_k. \quad (29)$$

By the mean-value theorem and Assumptions 1 and 2, we get

$$f(x_k + \lambda^{-1}\alpha_k d_k) - f(x_k) \leq \lambda^{-1}\alpha_k g_k^T d_k + L\lambda^{-2}\alpha_k^2 \|d_k\|^2. \quad (30)$$

Using this inequality with (18) and (29), we have

$$\alpha_k \geq -\frac{(1-\delta)\lambda}{L} \frac{g_k^T d_k}{\|d_k\|^2} = \frac{(1-\delta)\lambda}{L} \frac{\|g_k\|^2}{\|d_k\|^2}. \quad (31)$$

Letting $c = \min\{1, (1-\delta)\lambda/L\}$, we get (26), which completes the proof. $\square$

From inequalities (26) and (28), we can easily obtain the following lemma.

Lemma 3. *Suppose that Assumptions 1 and 2 hold. Let $\{x_k\}$ be generated by Algorithm MDL-CG where the line search satisfies the Armijo line search (22), then*

$$\sum_{k \geq 0} \frac{\|g_k\|^4}{\|d_k\|^2} < +\infty. \quad (32)$$

Next, we establish the global convergence theorem for Algorithm MDL-CG for uniformly convex functions.

Theorem 1. *Suppose that Assumptions 1 and 2 hold and f is uniformly convex on $\mathcal{L}$, namely, there exists a positive constant $\gamma > 0$ such that*

$$\gamma \|x - y\|^2 \leq (\nabla f(x) - \nabla f(y))^T (x - y), \quad \forall x, y \in \mathcal{L}. \quad (33)$$

Let $\{x_k\}$ and $\{d_k\}$ be generated by Algorithm MDL-CG and let α_k satisfy the Armijo condition (22), then one has either $g_k = 0$ for some k or

$$\lim_{k \rightarrow \infty} \|g_k\| = 0. \quad (34)$$

Proof. Suppose that $g_k \neq 0$ for all $k \geq 0$. By the convexity assumption (33), we have

$$d_{k-1}^T \tilde{y}_{k-1}^* \geq d_{k-1}^T y_{k-1} \geq \gamma \alpha_{k-1}^{-1} \|s_{k-1}\|^2. \quad (35)$$

Combining the previous relation with Lemma 1, we obtain

$$\begin{aligned} |\tilde{\beta}_k| &= \left| \frac{g_k^T (\tilde{y}_{k-1}^* - t s_{k-1})}{d_{k-1}^T \tilde{y}_{k-1}^*} \right| \leq \frac{\|g_k\| (\|\tilde{y}_{k-1}^*\| + t \|s_{k-1}\|)}{|d_{k-1}^T \tilde{y}_{k-1}^*|} \\ &\leq \frac{4L + t}{\gamma} \frac{\|g_k\|}{\|d_{k-1}\|}. \end{aligned} \quad (36)$$

Therefore, by the definition of the search direction d_k in (17) together with the previous inequality, we give an upper bound for $\|d_k\|$:

$$\begin{aligned} \|d_k\| &\leq \|g_k\| + |\tilde{\beta}_k| \frac{\|d_{k-1}\| \|g_k\|}{\|g_k\|^2} \|g_k\| + |\tilde{\beta}_k| \|d_{k-1}\| \\ &\leq \|g_k\| + 2 |\tilde{\beta}_k| \|d_{k-1}\| \\ &\leq \left(1 + 2 \frac{4L + t}{\gamma}\right) \|g_k\|. \end{aligned} \quad (37)$$

Inserting this upper bound for d_k in (32) yields $\sum_{k \geq 0} \|g_k\|^2 < \infty$, which completes the proof. $\square$

For simplicity, in Algorithm 1 in case the update parameter β_k is computed by (16), we refer it as Algorithm MDL⁺-CG. In the following, we show that Algorithm MDL⁺-CG is globally convergent for general nonlinear functions under the Wolfe line search conditions (23) and (24). In the rest of this section, we assume that convergence does not occur, which implies that there exists a positive constant $\mu > 0$ such that

$$\|g_k\| \geq \mu, \quad \forall k \geq 0. \quad (38)$$

Lemma 4. *Suppose that Assumptions 1 and 2 hold. Let $\{x_k\}$ and $\{d_k\}$ be generated by Algorithm MDL⁺-CG and let α_k be obtained by the Wolfe line search (23) and (24), then there exist positive constants C_1 and C_2 such that for all $k \geq 1$,*

$$|\tilde{\beta}_k^*| \leq C_1 \|s_{k-1}\|, \quad (39)$$

$$|\tilde{\beta}_k^*| \frac{|g_k^T d_{k-1}|}{\|g_k\|^2} \leq C_2 \|s_{k-1}\|. \quad (40)$$

Proof. From (14), (18) and (24), we have

$$\begin{aligned} d_{k-1}^T \tilde{y}_{k-1}^* &\geq d_{k-1}^T y_{k-1} \geq (\sigma_2 - 1) g_{k-1}^T d_{k-1} \\ &= (1 - \sigma_2) \|g_{k-1}\|^2. \end{aligned} \quad (41)$$

Utilizing this with Lemma 1, Assumption 2, and relations (21) and (38), we have

$$\begin{aligned} |\tilde{\beta}_k^*| &\leq \left| \frac{g_k^T \tilde{y}_{k-1}^*}{d_{k-1}^T \tilde{y}_{k-1}^*} \right| + t \left| \frac{g_k^T s_{k-1}}{d_{k-1}^T \tilde{y}_{k-1}^*} \right| \\ &\leq \frac{4ML\|s_{k-1}\|}{(1-\sigma_2)\mu^2} + t \frac{M\|s_{k-1}\|}{(1-\sigma_2)\mu^2} \\ &= \frac{(4L+t)M}{(1-\sigma_2)\mu^2} \|s_{k-1}\|. \end{aligned} \quad (42)$$

Letting $C_1 = (4L+t)M/(1-\sigma_2)\mu^2$, then (39) is satisfied. Furthermore, by the Wolfe condition (24), we have

$$g_k^T d_{k-1} \geq \sigma_2 g_{k-1}^T d_{k-1} \geq -\sigma_2 d_{k-1}^T \tilde{y}_{k-1}^* + \sigma_2 g_k^T d_{k-1}. \quad (43)$$

Also, observe that

$$g_k^T d_{k-1} = d_{k-1}^T y_{k-1} + g_{k-1}^T d_{k-1} \leq d_{k-1}^T \tilde{y}_{k-1}^*. \quad (44)$$

By rearranging inequality (43), we obtain $g_k^T d_{k-1} \geq -(\sigma_2/1 - \sigma_2)d_{k-1}^T \tilde{y}_{k-1}^*$, and together with (44), we obtain

$$\left| \frac{g_k^T d_{k-1}}{d_{k-1}^T \tilde{y}_{k-1}^*} \right| \leq \max \left\{ \frac{\sigma_2}{(1-\sigma_2)}, 1 \right\}. \quad (45)$$

It follows from Assumption 1 and (18), (24), (38), and (39) that

$$\begin{aligned} |\tilde{\beta}_k^*| \frac{|g_k^T d_{k-1}|}{\|g_k\|^2} &\leq \frac{\|g_k\|(\|\tilde{y}_{k-1}^*\| + t\|s_{k-1}\|)}{|d_{k-1}^T \tilde{y}_{k-1}^*|} \frac{|g_k^T d_{k-1}|}{\|g_k\|^2} \\ &\leq \frac{(4L+t)}{\mu} \max \left\{ \frac{\sigma_2}{(1-\sigma_2)}, 1 \right\} \|s_{k-1}\|. \end{aligned} \quad (46)$$

Letting $C_2 = (4L+t) \max\{\sigma_2/(1-\sigma_2), 1\}/\mu$, we obtain (40). $\square$

Next, we present a lemma which shows that, asymptotically, the search directions change slowly.

Lemma 5. Suppose that Assumptions 1 and 2 hold. Let $\{x_k\}$ and $\{d_k\}$ be generated by Algorithm MDL⁺-CG and let α_k be obtained by the Wolfe line search (23) and (24), then $d_k \neq 0$ and

$$\sum_{k \geq 1} \|w_k - w_{k-1}\|^2 < \infty, \quad (47)$$

where $w_k = d_k/\|d_k\|$.

Proof. Firstly, note that $d_k \neq 0$, for otherwise (18) would imply $g_k = 0$. Therefore, w_k is well defined. Next, we divide formula $\tilde{\beta}_k^*$ in two parts as follows:

$$\tilde{\beta}_k^* = \tilde{\beta}_k^{(1)} + \tilde{\beta}_k^{(2)}, \quad (48)$$

where

$$\begin{aligned} \tilde{\beta}_k^{(1)} &= \max \left\{ \frac{g_k^T \tilde{y}_{k-1}^*}{d_{k-1}^T \tilde{y}_{k-1}^*}, 0 \right\}, \\ \tilde{\beta}_k^{(2)} &= -t \frac{g_k^T s_{k-1}}{d_{k-1}^T \tilde{y}_{k-1}^*}. \end{aligned} \quad (49)$$

Moreover, let us define a vector r_k and a scalar δ_k by

$$r_k := \frac{v_k}{\|d_k\|}, \quad \delta_k := \beta_k^{(1)} \frac{\|d_{k-1}\|}{\|d_k\|}, \quad (50)$$

where

$$v_k = - \left(1 + \tilde{\beta}_k^* \frac{g_k^T d_{k-1}}{\|g_k\|^2} \right) g_k + \tilde{\beta}_k^{(2)} d_{k-1}. \quad (51)$$

Therefore, from (17), for $k \geq 1$, we obtain

$$w_k = r_k + \delta_k w_{k-1}. \quad (52)$$

Using this relation with the identity $\|w_k\| = \|w_{k-1}\| = 1$, we have that

$$\|r_k\| = \|w_k - \delta_k w_{k-1}\| = \|w_{k-1} - \delta_k w_k\|. \quad (53)$$

In addition, using this with the condition $\delta_k \geq 0$ and the triangle inequality, we get

$$\begin{aligned} \|w_k - w_{k-1}\| &\leq \|w_k - \delta_k w_{k-1}\| + \|w_{k-1} - \delta_k w_k\| \\ &= 2\|r_k\|. \end{aligned} \quad (54)$$

Now, we evaluate the quantity v_k . It follows from the definition of v_k in (52) and (21), (39), (40), and (45) that there exists a positive constant $D > 0$ such that

$$\begin{aligned} \|v_k\| &= \left\| - \left(1 + \tilde{\beta}_k^* \frac{g_k^T d_{k-1}}{\|g_k\|^2} \right) g_k + \tilde{\beta}_k^{(2)} d_{k-1} \right\| \\ &\leq \left(1 + |\tilde{\beta}_k^*| \frac{|g_k^T d_{k-1}|}{\|g_k\|^2} \right) \|g_k\| + t \left| \frac{g_k^T s_{k-1}}{d_{k-1}^T \tilde{y}_{k-1}^*} \right| \|d_{k-1}\| \\ &\leq \left(1 + |\tilde{\beta}_k^*| \frac{|g_k^T d_{k-1}|}{\|g_k\|^2} \right) \|g_k\| + t \left| \frac{g_k^T d_{k-1}}{d_{k-1}^T \tilde{y}_{k-1}^*} \right| \|s_{k-1}\| \\ &\leq (1 + C_2)M + t \max \left\{ \frac{\sigma_2}{(1-\sigma_2)}, 1 \right\} B \triangleq D. \end{aligned} \quad (55)$$

From the previous relation and Lemma 3, we obtain

$$\begin{aligned} \sum_{k \geq 0} \|r_k\|^2 &= \sum_{k \geq 0} \frac{\|v_k\|^2}{\|d_k\|^2} = \sum_{k \geq 0} \frac{\|v_k\|^2}{\|g_k\|^4} \frac{\|g_k\|^4}{\|d_k\|^2} \\ &\leq \frac{D^2}{\mu^4} \sum_{k \geq 0} \frac{\|g_k\|^4}{\|d_k\|^2} < +\infty. \end{aligned} \quad (56)$$

Therefore, using this with (54), we complete the proof. $\square$

Let Z^+ denote the set of positive integers. For $\lambda > 0$ and a positive integer Δ , we define the set

$$\mathcal{K}_{k,\Delta}^\lambda := \{i \in Z^+ \mid k \leq i \leq k + \Delta - 1, \|s_{i-1}\| > \lambda\}. \quad (57)$$

Let $|\mathcal{K}_{k,\Delta}^\lambda|$ denote the number of elements in $\mathcal{K}_{k,\Delta}^\lambda$. The following lemma shows that if the gradients are bounded away from zero and Lemma 4 holds, then a certain fraction of the steps cannot be too small. This lemma is equivalent to Lemma 3.5 in [10] and Lemma 4.2 in [28].

Lemma 6. *Suppose that all assumptions of Lemma 5 hold. Then, there exists constant $\lambda > 0$ such that, for any $\Delta \in Z^+$ and any index k_0 , there exists a greater index $k \geq k_0$ such that*

$$|\mathcal{K}_{k,\Delta}^\lambda| > \frac{\Delta}{2}. \quad (58)$$

Next, making use of Lemmas 4, 5, and 6, we can establish the global convergence theorem for Algorithm MDL⁺-CG under the Wolfe line search for general functions whose proof is similar to that of Theorem 3.6 in [10] and Theorem 4.3 in [28], thus we omit it.

Theorem 2. *Suppose that Assumptions 1 and 2 hold. If $\{x_k\}$ is obtained by Algorithm MDL⁺-CG and α_k is obtained by the Wolfe line search (23) and (24), then one has either $g_k = 0$ for some k or*

$$\liminf_{k \rightarrow \infty} \|g_k\| = 0. \quad (59)$$

4. Numerical Experiments

In this section, we report numerical experiments which were performed on a set of 73 unconstrained optimization problems. These test problems with the given initial points can be found in Andrei Neculai's web site (<http://camo.ici.ro/neculai/SCALCG/testuo.pdf>). Each test function made an experiment with the number of variables 1000, 5000, and 10000, respectively.

We evaluate the performance of our proposed conjugate gradient method MDL⁺-CG with that of the CG-DESCENT method [16]. The CG-DESCENT code is coauthored by Hager and Zhang obtained from Hager's web page (<http://www.math.ufl.edu/~hager/papers/CG/>). The implementation code was written in Fortran and have been compiled with the Intel Fortran compiler ifort (with compiler settings -O2 -double-size 128) on a PC (2.66 GHz Quad-Core processor, 4 Gbyte RAM) running Linux operating system. All algorithms were implemented with the Wolfe line search proposed by Hager and Zhang [16] and the parameters were set as default. In our experiments, the termination criterion is $\|g_k\| \leq 10^{-6}$ and set $u = s_{k-1}$ as in [11]. In the sequel, we focus our interest on our experimental analysis for the best value of parameter t ; hence, we have tested values of t ranging from 0 to 1 in steps of 0.005. The detailed numerical results can be found in the web site: http://www.math.upatras.gr/~livieris/Results/MDL_results.zip.

Figure 1 presents the percentage of the test problems that were successfully solved by Algorithm MDL⁺-CG for each

FIGURE 1: Percentage of successfully solved problems by Algorithm MDL⁺-CG for each value of parameter t .

FIGURE 2: Multigraph of means with respect to the function evaluations (blue line) and gradient evaluations (red line).

choice of parameter t and Figure 2 presents the multigraph of means with respect to function and gradient evaluations. Algorithm MDL⁺-CG reports the best results relative to the success rate for choices of t which belong to the interval $[0.785, 0.825]$. Moreover, in case $t = 0.82$, Algorithm MDL⁺-CG illustrates the highest success rate (98.64%), solving 216 out of 219 of the test problems successfully. Clearly, the interpretation in Figures 1 and 2 presents the influence of parameter t in the computational cost is more sensitive, hence, we focus our attention to the function and gradient evaluations metrics. Based on the experimental results performed on this limited test set we conjecture that the optimal parameter t with respect to the computational cost belongs to the interval $[0.02, 0.115]$. Notice that in case $t = 0.07$, Algorithm MDL⁺-CG illustrates the least mean number of function and gradient evaluations. Furthermore, for values of t in the intervals $[0.32, 0.42]$, $[0.61, 0.70]$, and $[0.95, 1]$, Algorithm MDL⁺-CG exhibits its worst performance with respect to the computational cost. It is worth noticing that

FIGURE 3: Performance profiles of CG-DESCENT, MDL_1^+ , MDL_2^+ and MDL_3^+ based on function evaluations.

FIGURE 4: Performance profiles of CG-DESCENT, MDL_1^+ , MDL_2^+ and MDL_3^+ based on gradient evaluations.

the choice $t = 0.995$ exhibits the worst performance in terms of computational cost and success rate.

We conclude our analysis by considering the performance profiles of Dolan and Moré [25] for the worst and the best parameter t choices. The use of performance profiles provide a wealth of information such as solver efficiency, robustness, and probability of success in compact form and eliminate the influence of a small number of problems on the benchmarking process and the sensitivity of results associated with the ranking of solvers [25]. The performance profile plots the fraction P of problems for which any given method is within a factor τ of the best method. The horizontal axis of the figure gives the percentage of the test

FIGURE 5: Performance profiles of CG-DESCENT, MDL_1^+ , MDL_2^+ and MDL_3^+ based on CPU time.

problems for which a method is the fastest (efficiency), while the vertical axis gives the percentage of the test problems that were successfully solved by each method (robustness). The curves in Figures 3 and 4 have the following meaning.

- (i) “CG-DESCENT” stands for the CG-DESCENT method.
- (ii) “ MDL_1^+ ” stands for Algorithm MDL^+ -CG with $t = 0.07$.
- (iii) “ MDL_2^+ ” stands for Algorithm MDL^+ -CG with $t = 0.82$.
- (iv) “ MDL_3^+ ” stands for Algorithm MDL^+ -CG with $t = 0.995$.

Figures 3–5 present the performance profiles of CG-DESCENT, MDL_1^+ , MDL_2^+ , and MDL_3^+ relative to the function evaluations, gradient evaluations, and CPU time (in seconds), respectively. Obviously, MDL_1^+ exhibits the best overall performance, significantly outperforming all other conjugate gradient methods, relative to all performance metrics. More analytically, MDL_1^+ solves about 64.4% and 66.2% of the test problems with the least number of function and gradient evaluations, respectively, while CG-DESCENT solves about 48.8% and 47%, in the same situations. Moreover, MDL_2^+ is more robust than the CG-DESCENT since it solves 55.3% and 53% of the test problems with the least number of function and gradient evaluations, respectively. As regarding the CPU time metric, the interpretation in Figure 5 illustrates that MDL_1^+ reports the best performance, followed by MDL_2^+ . More specifically, MDL_1^+ solves 68.4% of the test problems with the least CPU time while MDL_2^+ solves about 58% of the test problems. In terms of efficiency, MDL_2^+ and CG-DESCENT exhibit the best performance, successfully solving 216 out of 219 of the test problems. MDL_3^+ presents the worst performance, since its curves lie under the curves of the other conjugate gradient methods,

regarding all performance metrics. In summary, based on the performance of MDL_1^+ , MDL_2^+ and MDL_3^+ , we point out that the choice of parameter t crucially affects the efficiency of Algorithm MDL^+ -CG.

5. Conclusions and Future Research

In this paper, we proposed a conjugate gradient method which consists of a modification of Dai and Liao method. An important property of our proposed method is that it ensures sufficient descent independence of the accuracy of the line search. Moreover, it achieves a high-order accuracy in approximating the second-order curvature information of the objective function by utilizing the modified secant condition proposed by Babaie-Kafaki et al. [11]. Under mild conditions, we establish that the proposed method is globally convergent for general functions under the Wolfe line search conditions.

The preliminary numerical results show that if we choose a good value of parameter t , our proposed algorithm performs very well. However, we have not theoretically established an optimal parameter t , yet which consists our motivation for future research. Moreover, an interesting idea is to apply our proposed method to a variety of challenging real-world problems such as protein folding problems [29].

References

- [1] M. R. Hestenes and E. Stiefel, "Methods of conjugate gradients for solving linear systems," *Journal of Research of the National Bureau of Standards*, vol. 49, pp. 409–436, 1952.
- [2] R. Fletcher and C. M. Reeves, "Function minimization by conjugate gradients," *The Computer Journal*, vol. 7, pp. 149–154, 1964.
- [3] E. Polak and G. Ribière, "Note sur la convergence de méthodes de directions conjuguées," *Revue Française d'Informatique et de Recherche Operationnelle*, vol. 3, no. 16, pp. 35–43, 1969.
- [4] Y. Liu and C. Storey, "Efficient generalized conjugate gradient algorithms. I. Theory," *Journal of Optimization Theory and Applications*, vol. 69, no. 1, pp. 129–137, 1991.
- [5] Y. H. Dai and Y. Yuan, "A nonlinear conjugate gradient method with a strong global convergence property," *SIAM Journal on Optimization*, vol. 10, no. 1, pp. 177–182, 1999.
- [6] R. Fletcher, *Practical Methods of Optimization*, John Wiley & Sons, New York, NY, USA, 2nd edition, 1987.
- [7] Y. H. Dai and Y. X. Yuan, *Nonlinear Conjugate Gradient Methods*, Shanghai Scientific and Technical, Shanghai, China, 2000.
- [8] J. Nocedal and S. J. Wright, *Numerical Optimization*, Springer, New York, NY, USA, 1999.
- [9] W. W. Hager and H. Zhang, "A survey of nonlinear conjugate gradient methods," *Pacific Journal of Optimization*, vol. 2, no. 1, pp. 35–58, 2006.
- [10] Y. H. Dai and L. Z. Liao, "New conjugacy conditions and related nonlinear conjugate gradient methods," *Applied Mathematics and Optimization*, vol. 43, no. 1, pp. 87–101, 2001.
- [11] S. Babaie-Kafaki, R. Ghanbari, and N. Mahdavi-Amiri, "Two new conjugate gradient methods based on modified secant equations," *Journal of Computational and Applied Mathematics*, vol. 234, no. 5, pp. 1374–1386, 2010.
- [12] J. A. Ford, Y. Narushima, and H. Yabe, "Multi-step nonlinear conjugate gradient methods for unconstrained minimization," *Computational Optimization and Applications*, vol. 40, no. 2, pp. 191–216, 2008.
- [13] G. Li, C. Tang, and Z. Wei, "New conjugacy condition and related new conjugate gradient methods for unconstrained optimization," *Journal of Computational and Applied Mathematics*, vol. 202, no. 2, pp. 523–539, 2007.
- [14] W. Zhou and L. Zhang, "A nonlinear conjugate gradient method based on the MBFGS secant condition," *Optimization Methods & Software*, vol. 21, no. 5, pp. 707–714, 2006.
- [15] H. Yabe and M. Takano, "Global convergence properties of nonlinear conjugate gradient methods with modified secant condition," *Computational Optimization and Applications*, vol. 28, no. 2, pp. 203–225, 2004.
- [16] W. W. Hager and H. Zhang, "A new conjugate gradient method with guaranteed descent and an efficient line search," *SIAM Journal on Optimization*, vol. 16, no. 1, pp. 170–192, 2005.
- [17] L. Zhang, W. Zhou, and D. Li, "Global convergence of a modified Fletcher-Reeves conjugate gradient method with Armijo-type line search," *Numerische Mathematik*, vol. 104, no. 4, pp. 561–572, 2006.
- [18] W. Cheng and Q. Liu, "Sufficient descent nonlinear conjugate gradient methods with conjugacy condition," *Numerical Algorithms*, vol. 53, no. 1, pp. 113–131, 2010.
- [19] Z. Dai and B. S. Tian, "Global convergence of some modified PRP nonlinear conjugate gradient methods," *Optimization Letters*, vol. 5, no. 4, pp. 615–630, 2011.
- [20] S. Q. Du and Y. Y. Chen, "Global convergence of a modified spectral FR conjugate gradient method," *Applied Mathematics and Computation*, vol. 202, no. 2, pp. 766–770, 2008.
- [21] A. Lu, H. Liu, X. Zheng, and W. Cong, "A variant spectral-type FR conjugate gradient method and its global convergence," *Applied Mathematics and Computation*, vol. 217, no. 12, pp. 5547–5552, 2011.
- [22] L. Zhang, "Two modified Dai-Yuan nonlinear conjugate gradient methods," *Numerical Algorithms*, vol. 50, no. 1, pp. 1–16, 2009.
- [23] L. Zhang, "New versions of the Hestenes-Stiefel nonlinear conjugate gradient method based on the secant condition for optimization," *Computational & Applied Mathematics*, vol. 28, no. 1, pp. 111–133, 2009.
- [24] L. Zhang and W. Zhou, "Two descent hybrid conjugate gradient methods for optimization," *Journal of Computational and Applied Mathematics*, vol. 216, no. 1, pp. 251–264, 2008.
- [25] E. D. Dolan and J. J. Moré, "Benchmarking optimization software with performance profiles," *Mathematical Programming*, vol. 91, no. 2, pp. 201–213, 2002.
- [26] 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*, vol. 102, no. 1, pp. 147–167, 1999.
- [27] J. Zhang and C. Xu, "Properties and numerical performance of quasi-Newton methods with modified quasi-Newton equations," *Journal of Computational and Applied Mathematics*, vol. 137, no. 2, pp. 269–278, 2001.
- [28] J. C. Gilbert and J. Nocedal, "Global convergence properties of conjugate gradient methods for optimization," *SIAM Journal on Optimization*, vol. 2, no. 1, pp. 21–42, 1992.
- [29] J. D. Bryngelson, J. N. Onuchic, N. D. Socci, and P. G. Wolynes, "Funnels, pathways, and the energy landscape of protein folding: a synthesis," *Proteins*, vol. 21, no. 3, pp. 167–195, 1995.