BISKA

New Trends in Mathematical Sciences

New adaptive conjugate gradient methods choices for unconstrained optimization

Hawraz N. Jabbar

Department of mathematics, Kirkuk university, Kirkuk, Iraq

Abstract: In this paper, we present two conjugate gradient methods choices for solving unconstrained optimization problems. This attempts is to find suitable choices for parameter of a nonlinear conjugate gradient method proposed by Dai and Liao based on the matrix analysis and using the memoryless BFGS updating formula. Numerical results show that the proposed method is efficient for the unconstrained problems in the CUTEr collection.

Keywords: Unconstrained optimization, conjugate gradient method, BFGS update, line search, condition number, global convergence.

1 Introduction

Conjugate gradient methods are a powerful line search method for solving large-scale optimization problems due to their simplicity and low memory storage. The first CG-method was proposed by Hestenes and Stiefel [1] to solve the system of linear equations, while a nonlinear method was introduced by Fletcher and Reeves [2] . Conjugate gradient (CG) methods have attracted special attention for solving large-scale unconstrained optimization problems in the form.

$$min_{x \in R^n} f(x) \tag{1}$$

the objective function $f : R \to R^n$ is supposed to be continuously differentiable, bounded from below, The method generate a sequence of iterates.

$$x_{k+1} = x_k + s_k, \quad k = 0, 1, 2 \tag{2}$$

where $s_k = \alpha_k d_k$, α_k is a suitable step length, and d_k is a search direction computed recursively by:

$$d_0 = -g_0, d_{k+1} = -g_{k+1} + \beta_k d_k \quad k = 0, 1, 2 \tag{3}$$

where $g_k = \nabla f(x_k) is$ the gradient of the objective function, and β_k is a real scalar called conjugate parameter. The good features of the method attracted many authors to develop it, and to design algorithms for solving optimization problems. see the works by Polak and Ribière [3] , Polyak [4] , Dai and Liao [5] , Hager and Zhang [6,7] , Babaie-Kafaki and Ghanbari [8,9,10,11,13], Babaie-Kafaki [12] and Andrei [14,15].

Usually, the step size α_k is computed to satisfy some line search conditions [16]. In the convergence analyses and implementation of conjugate gradient algorithms the standard Wolfe conditions [17,18]

$$f(x_k + \alpha_k d_k) - f(x_k) \leq \mu \alpha_k \, g_k^T d_k \tag{4}$$

* Corresponding author e-mail: hawrazmath@gmail.com

$$g_{k+1}^T d_k \geq \sigma g_k^T d_k \tag{5}$$

where d_k is a descent direction and $\phi < \mu < \sigma < 1$, often have been considered. Also, the strong Wolfe line search conditions consisting of (4) and can be used.

$$|g_{k+1}^T d_k| \leq -\sigma g_k^T d_k \tag{6}$$

In an attempt to use quasi-Newton techniques in conjugate gradient algorithms, one of the essential CG parameters has been proposed by Dai and Liao [5], can be written in the form.

$$\beta_k^{DL} = \frac{g_{k+1}^T y_k}{d_k^T y_k} - t \frac{g_{k+1}^T s_k}{d_k^T y_k} \tag{7}$$

where $y_k = g_{k+1} - g_k$, and t is a nonnegative parameter which has considerable effect on numerical performance of the DL method. For general objective functions Dai and Liao proposed a new version for DL by using Powell's nonnegative restriction [20], that given by

$$\beta_k^{DL^+} = max\{\frac{g_{k+1}^T y_k}{d_k^T y_k}, 0\} - t \frac{g_{k+1}^T s_k}{d_k^T y_k} \tag{8}$$

Dai-Liao conjugate gradient method is one of the most efficient conjugate method, and is has been widely extended by optimizing the parameter t in different ways. The search directions (7) of the DL method and (3) can be written as:

$$d_0 = -g_0, \ d_{k+1} = -Q_{k+1} g_{k+1} \ \ k = 0, 1, 2 \tag{9}$$

in which

$$Q_{k+1} = I - \frac{s_k y_k^T}{y_k^T s_k} + t \frac{s_k s_k^T}{y_k^T s_k} \tag{10}$$

is called the search direction matrix, and the above matrix (9) may not be symmetric and positive definite, therefore, the direction generated by (8) does not satisfy the typical quasi-Newton condition. But, the direction (7) satisfies Dai-Liao conjugate condition.

$$d_{k+1}^T y_k = -t \ g_{k+1}^T s_k \tag{11}$$

widely extended by optimizing the parameter t in different ways, Hager and Zhang [6] presented the following choice for parameter β_k.

$$\beta_k^{CGHZ} = max\{\beta_K^{HZ} \frac{-1}{\|d_k\| min\{\eta, \|g_k\|\}}\} \tag{12}$$

where

$$\beta_k^{HZ} = \frac{g_{k+1}^T y_k}{d_k^T y_k} - 2 \frac{\|y_k\|^2}{d_k^T y_k} \frac{g_{k+1}^T s_k}{d_k^T y_k} \tag{13}$$

The parameter β_k^{HZ} is one of the special case of β_k^{DL} with $t = 2\frac{\|y_k\|^2}{d_k^T y_k}$. Then, minimizing some different upper bounds of the spectral and Frobenius condition number of the search direction matrix Q_{k+1}, In [8] saman and Gambari, proposed two choices for the parameter t :

$$t_{k_1}^* = \frac{s_k^T y_k}{\|s_k\|^2} + \frac{\|y_k\|}{\|s_k\|}, \qquad t_{k_2}^* = \frac{\|y_k\|}{\|s_k\|} \tag{14}$$

where $\|.\|$ stands for the Euclidean norm. Here, we propose a new two adaptive choices for t in (10), using the memoryless BFGS updating formula [19]. The structure of the paper is as follows. In Section 2, based on a matrix analysis.

2 Two new adaptive choices for (DL) Dai and Liao

In this part, we present the new adaptive choices for DL for the parameter t in (10) by minimizing the condition number.

2.1 The first adaptive choice for DL:

Here, we suggest the first adaptive choice in the form

$$t_1 = \zeta(1 + \frac{\|y_k\|^2}{s_k^T y_k}) \tag{15}$$

where $\zeta > 0$ is real parameter. If $\zeta = 1$then this makes the search direction matrix Q_{k+1} given by (10) to be similar to the memoryless BFGS updating formula H_{k+1} given by in the form

$$H_{k+1} = I - \frac{s_k y_k^T + y_k s_k^T}{s_k^T y_k} + (1 + \frac{\|y_k\|^2}{s_k^T y_k})\frac{s_k s_k^T}{s_k^T y_k} \tag{16}$$

Where H_{k+1} is an approximation for $\nabla^2 f(x_{k+1})$. Babaie-Kafaki and Ghanbari in [8], show that the matrix Q_{k+1} given by (10) has $n-2$ eigenvalues equal to 1, and two other singular values ∂_k^+ and ∂_k^- satisfying

$$\partial_k^- \partial_k^+ = t\frac{\|s_k\|^2}{s_k^T y_k} \tag{17}$$

$$\partial_k^{-2} \partial_k^{+2} = t^2 \frac{\|s_k\|^4}{(s_k^T y_k)^2} + \frac{\|s_k\|^2 \|y_k\|^2}{(s_k^T y_k)^2} \tag{18}$$

where $\partial_k^- \geq 1 \geq \partial_k^+$, and we can suggest the following upper bound for the spectral condition number of Q_{k+1}:

$$k(Q_{k+1}) = \frac{\partial_k^+}{\partial_k^-} \leq \frac{\partial_k^{-2} + \partial_k^{+2}}{\partial_k^- \partial_k^+} \tag{19}$$

Now, from (17), (18) and (15), we get

$$\partial_k^- \partial_k^+ = \zeta(1 + \frac{\|y_k\|^2}{s_k^T y_k})\frac{\|s_k\|^2}{s_k^T y_k} = \zeta\frac{\|s_k\|^2}{s_k^T y_k} + \zeta\frac{\|y_k\|^2}{s_k^T y_k}\frac{\|s_k\|^2}{s_k^T y_k} \tag{20}$$

and

$$\partial_k^{-2} + \partial_k^{+2} = \zeta^2(1 + \frac{\|y_k\|^4}{s_k^T y_k})^2\frac{\|s_k\|^4}{(s_k^T y_k)^2} + \frac{\|y_k\|^2 \|s_k\|^2}{(s_k^T y_k)^2} \tag{21}$$

we put (20) and (21) in (19), we obtain

$$k(Q_{k+1}) \leq \zeta(1 + \frac{\|y_k\|^2}{s_k^T y_k})\frac{\|s_k\|^2}{s_k^T y_k} + \frac{1}{\zeta}\frac{\|y_k\|^2}{\|y_k\|^2 + s_k^T y_k} \tag{22}$$

The unique minimizer of the upper bound of $k(Q_{k+1})$ given by (22), can be written

$$\zeta_k^{H1} = \frac{\|y_k\|}{\|s_k\|}\frac{s_k^T y_k}{s_k^T y_k + \|y_k\|^2} \tag{23}$$

Now, from (23) and (15), we get

$$t_k^{H1} = t_1 = \frac{\|y_k\|}{\|s_k\|} \frac{s_k^T y_k}{s_k^T y_k + \|y_k\|^2}(1 + \frac{\|y_k\|^2}{s_k^T y_k}) \tag{24}$$

2.2 The second adaptive choice for DL:

We suggest the second adaptive choice in the form

$$t_2 = (1 + \zeta)\frac{\|y_k\|^2}{s_k^T y_k} \tag{25}$$

where $\zeta > 0$ is real parameter. If $\zeta = 1$, then this makes the search direction matrix Q_{k+1} given by (10) to be similar to the search direction that proposed by Hager and Zhang in equation (13),If $\zeta = 0$, then this makes the search direction matrix Q_{k+1} given by (10) to be similar to the search direction that given by (16).

Now, from (17), (18) and (25), we get

$$\partial_k^- \partial_k^+ = (1 + \zeta)\frac{\|y_k\|^2}{s_k^T y_k}\frac{\|s_k\|^2}{s_k^T y_k} = \frac{\|y_k\|^2}{s_k^T y_k}\frac{\|s_k\|^2}{s_k^T y_k} + \zeta\frac{\|y_k\|^2}{s_k^T y_k}\frac{\|s_k\|^2}{s_k^T y_k} \geq \zeta\frac{\|y_k\|^2}{s_k^T y_k}\frac{\|s_k\|^2}{s_k^T y_k} \tag{26}$$

and

$$\partial_k^{-2} + \partial_k^{+2} = (1 + \zeta)^2\frac{\|y_k\|^4\|s_k\|^4}{(s_k^T y_k)^4} + \frac{\|y_k\|^2\|s_k\|^2}{(s_k^T y_k)^2} = \zeta^2(1 + \frac{\|y_k\|^2}{s_k^T y_k})^2\frac{\|s_k\|^4}{(s_k^T y_k)^2} + \frac{\|y_k\|^2\|s_k\|^2}{(s_k^T y_k)^2} \tag{27}$$

we put (26) and (27) in (19), we obtain

$$k(Q_{k+1}) \leq \zeta\frac{\|y_k\|^2\|s_k\|^2}{(s_k^T y_k)^2} + \frac{1}{\zeta}\frac{\|y_k\|^2\|s_k\|^2 + (s_k^T y_k)^2}{(s_k^T y_k)^2} + c_1 \tag{28}$$

Where c_1 is a constant and an independent of ζ. After some algebraic manipulations, we can get

$$\zeta_k^{H2} = \sqrt{1 + \frac{(s_k^T y_k)^2}{\|s_k\|^2\|y_k\|^2}} \tag{29}$$

Is the unique minimizer of the upper bound of the $k(Q_{k+1})$ that given by (28) Now, we put (29) in (25), it can be seen that

$$t_k^{H2} = t_2 = [1 + \sqrt{1 + \frac{(s_k^T y_k)^2}{\|y_k\|^2\|s_k\|^2}}]\frac{\|y_k\|^2}{s_k^T y_k}. \tag{30}$$

3 Convergence analysis

In this part, we discuss the global convergence of the DL method by using the choices (24) and (30) for the parameter t. We need to make the following standard assumptions on the objective function.

3.1 Assumptions

(i)The objective function is bounded on the bounded level set $S = \{x \in R^n : f(x) \leq f(x_1)\}$.
(ii) In some neighborhood N of S, is continuously differentiable and its gradient is Lipschitz continuous i.e. $\exists$ a constant

$0 < L < 1$ such that

$$\|g(x) - g(y)\| \leq L\|x - y\| \quad \forall \, x, y \in N \tag{31}$$

is satisfies.

Under the above assumptions on f, there exist a constant $\gamma > 0$ such that

$$\|g(x)\| \leq \gamma, \quad for \; all \; x \in S \tag{32}$$

Now, if the search direction d_k is descent in the iterative method (2) with standard line search (4), then $\{x_k\}_{k \geq 0} \subseteq S$. Based on eigenvalue analysis, Babaie-Kafaki and Ghanbari [10] show that if

$$t > \frac{1}{4}\left(\frac{\|y_k\|^2}{s_k^T y_k} - \frac{s_k^T y_k}{\|s_k\|^2}\right), \tag{33}$$

then d_k of the DL method satisfy the descent condition. Directly, we seen, both t_1 given by (24) and t_2 given by (30) satisfies (33). Therefore, the DL method with $t = t_1, t_2$ owns the descent condition. The uniform convexity of the objective function f ensures that there exists a positive constant μ such that

$$s_k^T y_k \geq \mu \|s_k\|^2, \tag{34}$$

As seen, the equation (34) which together with (31) yields

$$t_k^{H1} = t_1 \leq \frac{L\mu}{\mu + L^2}\left(1 + \frac{L^2}{\mu}\right), \tag{35}$$

and

$$t_k^{H2} = t_2 \leq \left(1 + \sqrt{1 + \frac{\mu^2}{L^2}}\right)\frac{L^2}{\mu} \tag{36}$$

thus, $t = t_1, t_2$ are bounded for uniformly convex objective function. In the other hand it's consequently if the step size α_k is computed to satisfy some line search conditions (strong Wolf condition) (4) and (6), then the theorem 3.3 in [5], achieve the global convergence of the DL method with (24) and (30) for general objective function, even with uniform objective function.

4 Numerical results

The numerical results of the two new adaptive with $t = t_1, \; t_2$ are displayed in this section. We selected a number of 75 large-scale unconstrained optimization test functions in generalized or extended form, presented in [21], where the vast majority of problems are taken from CUTEr collection [22]. For each test function we have considered 10 numerical experiments with the number of variables increasing as n = 100, 200, ..., 1000. Therefore, the numerical experiments include a set of 750 unconstrained optimization test functions of different structures and complexities with the stopping condition $\|\nabla f(x_k)\|_\infty \leq 10^{-6}$. The new algorithms were implemented Fortran 77 applying to DL+ method with the different formulas (14), (24) and (30). so we denote the two new proposed choices in (24) and (30) by DLT1 and DLT2 respectively, we compare the new algorithms (DLT1 and DLT2) versus DLK1 and DLK2 that showed in (14). All codes are written in Fortran 77 on PC, Intel(R) Core(TM) 2 Duo CPU T6400 @ 2.00GHz (2 CPUs), RAM 4.00 GB. Fig. 1 presents the Dolan and Moré's [23] performance profile of DLT1 and DLT2 versus DLK1 DLK2 in number of iterations.

Fig. 2 show that the performance profile of DLT1 and DLT2 versus DLK1 DLK2 in number of function and gradient evaluation. Fig. 3 show that the performance profile of DLT1 and DLT2 versus DLK1 DLK2 in the CPU time.
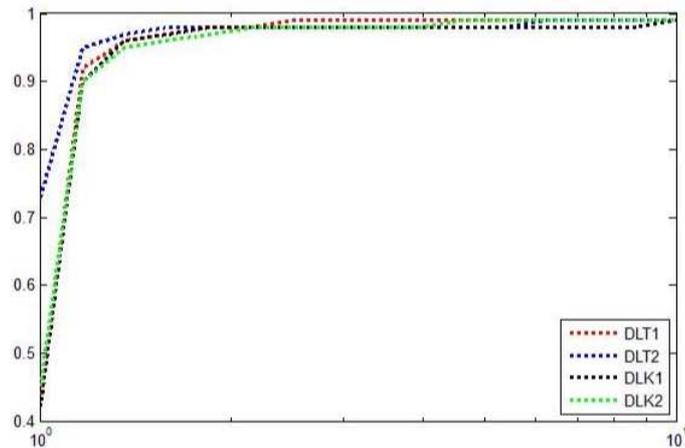

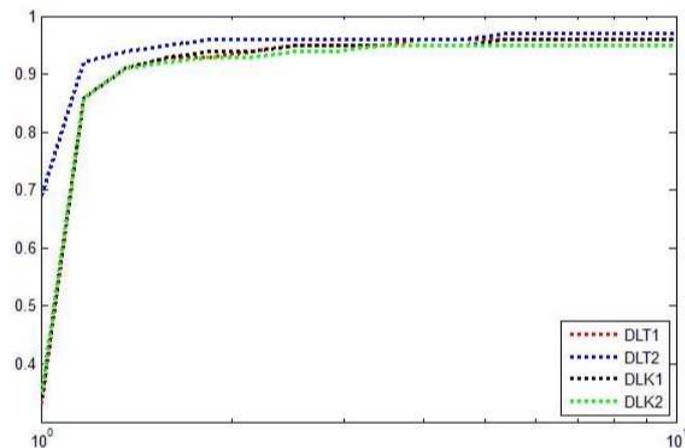**Fig. 1:** Total number of iteration performance profiles

Fig. 2: Total number of function and gradient evaluation performance profiles

5 Conclusion

The new algorithms presented a new choices of DL^+ conjugate gradient with descent property and global convergence with the Wolfe line search. The new algorithms DLT1 and DLT2 give a good result compares with the other algorithms.

Competing interests

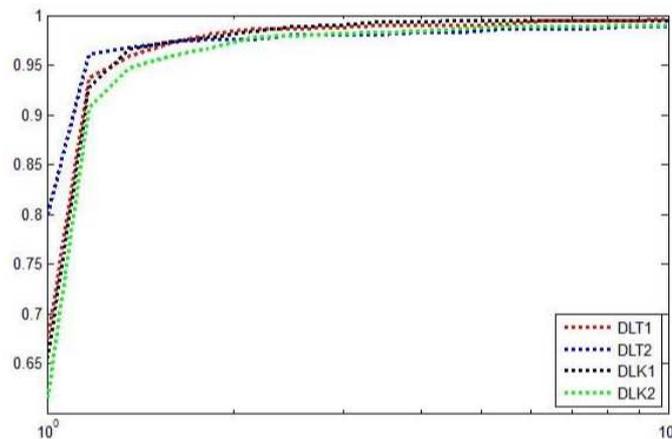The authors declare that they have no competing interests.

Fig. 3: CPU time performance profiless

Authors' contributions

All authors have contributed to all parts of the article. All authors read and approved the final manuscript.

References

[1] M.R. Hestenes and E. Stiefel, Methods of conjugate gradients for solving linear systems, J. Res. Nat. Bur. Stand.49 (1952), pp. 409-436.

[2] R. Fletcher and C.M. Reeves, Function minimization by conjugate gradients, Comput. J. 7 (1964), pp. 149-154.

[3] E. Polak and G. Ribière, Not sur la convergence de methods directions conjugees, Revue Franccaise d'Informatiqueet de Recherche opiérationnelle 16 (1969), pp. 35-43.

[4] B.T. Polyak, The conjugate gradient method in extremal problems, USSR Comp. Math. Phys. 9 (1969), pp. 94-112.

[5] Y.-H. Dai and L.Z. Liao, New conjugacy conditions and related nonlinear conjugate gradient methods, Appl. Math. Optim. 43 (2001), pp. 87-101.

[6] W.W. Hager and H. Zhang, A new conjugate gradient method with guaranteed descent and an efficient line search, SIAM J. Optim. 16 (2005), pp. 170-192.

[7] W.W. Hager and H. Zhang, The limited memory conjugate gradient method, SIAM J. Optim. 23(4) (2013), pp. 2150-2168.

[8] Babaie-Kafaki S, Ghanbari R (2014a), The Dai-Liao nonlinear conjugate gradient method with optimal parameter choices. Eur J Oper Res 234(3):625-630

[9] Babaie-Kafaki S, Ghanbari R (2014b), A descent extension of the Polak-Ribie 're-Polyak conjugate gradient method. Comput Math Appl 68(12):2005-2011

[10] Babaie-Kafaki S, Ghanbari R (2014c), A descent family of Dai-Liao conjugate gradient methods. Optim Methods Softw 29(3):583-591.

[11] Babaie-Kafaki S, Ghanbari R (2015), Two optimal Dai-Liao conjugate gradient methods. Optimization 64(11):2277-2287.

[12] Babaie-Kafaki S (2016), On optimality of two adaptive choices for the parameter of Dai-Liao method. Optim Lett 10:1789-1797.

[13] Babaie-Kafaki S, Ghanbari R (2017), Two Adaptive Dai-Liao Nonlinear Conjugate Gradient Methods. Iran J Sci Technol Trans Sci.

[14] N. Andrei, An adaptive conjugate gradient algorithm for large-scale unconstrained optimization. Journal of Computational and Applied Mathematics 292 (2016) 83-91.

[15] N. Andrei, Accelerated adaptive Perry conjugate gradient algorithms based on the self-scaling memoryless BFGS update. Journal of Computational and Applied Mathematics (2017)

[16] N. Andrei, Criticism of the Unconstrained Optimization Algorithms Reasoning. Editura Academiei Române, Bucureşti, (2009).

[17] P. Wolfe, Convergence conditions for ascent methods. SIAM Review 11 (1969) 226-235.

[18] P. Wolfe, Convergence conditions for ascent methods. II: Some corrections. SIAM Review 13 (1971) 185-188.

[19] Sun W, Yuan YX (2006), Optimization theory and methods: nonlinear programming. Springer, New York.

[20] Powell MJD (1984), Non-convex minimization calculations and the conjugate gradient method. In: Griffith DF. (ed) Numerical analysis (Dundee, 1983) volume 1066 of Lecture Notes in Math. Springer, Berlin, pp 122-141.

[21] Andrei,N., An unconstrained optimization test functions collection.Adv.Model.Optim. 10,147-161(2008).

[22] Bongartz, I., Conn, A.R., Gould, N.I.M., Toint, Ph.L.: CUTEr: constrained and unconstrained testing environments. ACM Trans. Math. Softw. 21, 123-160 (1995).

[23] Dolan ED, More ´ JJ (2002), Benchmarking optimization software with performance proles. Math Program 91(2,Ser. A):201-213.