A new pair conjugate algorithm with an adaptive optimal step size

Muna M. Mohammed Ali
Department of Mathematics
College of Computers sciences and Mathematics
Mosul University

Maha S. Y. Al-Salih
Department of Computer
College of Education
Mosul University

Abbas Y. Al-Bayati
Department of Mathematics
College of Computers sciences and Mathematics
Mosul University

Received 25 / 06 / 2008 **Accepted** 08 / 01 / 2009

الملخص

في هذا البحث تم استخدام خطوة مثلى ملائمة جديدة لزيادة كفاءة خوارزمية المترافقات المزدوجة القياسية. ففي خط البحث تم استخدام مقبولة من خلال الاستكمال التربيعي كخطوة بدائية لهذا الاستكمال مع التأكيد على حاجة التقنية الجديدة لحساب مشتقة واحدة في كل خطوة تكرارية ، تم أيضاً تطوير خوارزمية Armijo واستخدامها مع التقنية الجديدة . النتائج العددية اثبت كفاءة الطريقة بأستخدام التقنية المقترحة.

ABSTRACT

In this paper, a new adapted optimal step-size is designed to improve the efficiency of pair conjugate method. At each linear search an acceptable step- size is estimated during quadratic interpolation and this estimate is used as an initial trial step-size. The technique needs only additional gradient evaluation at each search direction. Also we have improved Armijo line search technique to be used together with the new optimal stepsize.

The numerical results are more efficient than the results of the same method using the classical scheme for the linear search technique.

1. Introduction:

The problem to be considered is that of finding a local minimum of a function *f(x)* of *n* variables $x=(x_1,\dots,x_n)$.

The gradient of *f* will be denoted by $g_i(x)$, and the matrix of second derivatives by G_{ij}. The function assumed to be differentiable so that g=0 at the minimum can be used. Methods will be discussed in this paper are all iterative and x_1,x_2... will be used to denote successive approximations to the minimum. Namely, all the methods are based or the iteration

$$x_{i+1} = x_i + \lambda_i d_i \tag{1}$$

The parameter λ_i is usually chosen to minimize $f(x_i+\lambda_i d_i)$ to make $f_{i+1} \le f_i$. But if λ_i minimizes f in this particular direction on obvious advantage of this approach is that each step process to converge faster see All-Baali(1985). The procedure for determining such λ_i sometimes called a step-length algorithms, so as to minimize

$$\phi(\lambda) = f(x_i+\lambda_i d_i), \tag{2}$$

is referred to as a line search procedure.

A number of numerical techniques for carrying out such line searches have been developed. Quadratic interpolation has been used eg. Gill and Murray(1974); and Scales,(1985), Raydan(1997),the technique needs function evaluations only and also cubic interpolation. Dixon(1972) which needs function and gradient evaluations. In the later category is the method of Davidon(1959) which is described in (Bunday, 1984) and has been used in the new programs for implementation the standard air conjugate algorithms in this paper.

In this paper, we present a new adaptive optimal step-size technique to improve the efficiency of the standard pair conjugate method, and replacing the standard cubic line search. The numerical results show that the new scheme is more efficient than the cubic search when using both with two pair conjugate methods to find the optimal step-size for solving several unconstrained test problems.

2. Implementation:

Before presenting the new technique, we shall derive the optimal step-size formula which is a given by:

$$\lambda_i = -\frac{g_i^T g_i}{g_i^T G d_i} \tag{3}$$

where λ_i is exact step-length used usually with the conjugate Gradient methods, See Bazaraa, (2000).

2.1 The derivation of the New λ_i :

Let f(x) be positive definite quadratic function as: $f(x)=\frac{1}{2}x^T Gx-b^T x+c$ where G is symmetric positive definite matrix. The gradient of $f(x)$ can be expressed in the ith iteration: $g_i=Gx_i-b$, the required minimum point along the line is $x_{i+1}=x_i+\lambda_i d_i$, where d_i is the search direction. Now, from the exact search property the following condition $d_i^T g^*=0\, or -g_i^T g_i^*=0$ (since $d_i=-g_i$) must hold (g* is the gradient at x*).

We have:

$$g^*=Gx^*-b$$
$$=G(x_i+\lambda_i d_i)-b$$
$$=Gx_i-b+\lambda_i d_i^T G$$
$$=g_i+\lambda_i d_i^T G$$

The minimum will be attained when

$$-g_i^T g_i^*=0$$
$$\Rightarrow 0=-g_i^T g_i-\lambda_i g_i^T Gd_i$$
$$\Rightarrow \lambda_i=-\frac{g_i^T g_i}{g_i^T Gd_i}$$

This is the optimal step-size parameter for the cases that ELS and descent conditions are satisfy.

3. Pair Conjugate Method:

Stewart (1977) introduced a generalization of the notion of conjugancy, leading to a variety of finitely terminating iterations for solving systems of linear equations. An adaptation of Steward's ideas to minimization problems confirms not only the above-mentioned suspicion, but establishes a method with an even wider scope of generality.

We note that the definition of conjugancy can also be phrased as follows: If the vectors $u_0,u_1,...,u_{n-1}$ are the columns of an $n\times n$ matrix$V$, then $u_0,u_1,...,u_{n-1}$ are A-conjugate if $U^T AU$ is a diagonal (and of course nonsingular). The generalization is achieved by introducing a second set of vectors $v_0,v_1,...,v_{n-1}$.

Definition 3.1

let A, U, and V be non singular $n\times n$ matrices. Then (U,V) is a pair G-conjugate if $L=U^T AV$ is lower triangular.

The generalized algorithm for solving the equations

$Gx + b = 0$

$g_{k-1} = Gx_{i-1} + b$

$\mu_{i-1} = -v_{i-1}^T g_{i-1} / v_{i-1}^T G u_{i-1}$

$x_{i-1} = x_{i-1} + \mu_{i-1} u_{i-1}$

Where $i = 1,2,...,n$, and where

$U = [u_0, ..., u_{n-1}]$ and $V = [v_0, ..., v_{n-1}]$.

Stewart (1977) developed an algorithm for constructing a pair G-conjugate pair *(U,V)* as follows. Given nonsingular matrices V,G and P, the vector u_k is determined as a linear combination of $p_o, p_i, ..., p_k$, $i = 0,1,...,n-1$, such that U and V are G-conjugate. The resulting algorithm is as follows:

$u_0 = d_0 p_0$,

$u_1 = d_1 [,[\; p_1 - (v_0^T G p_1 / v_0^T G u_0) u_0$

$\vdots$.

$\vdots$

$u_k = d_k [,[\; p_k - (v_0^T G p_k / v_0^T G u_0) u_0 - (v_1^T G p_k / v_1^T G u_1) u_1 - \cdots - (v_{k-1}^T G p_k / v_{k-1}^T G u_{k-1}) u_{k-1}$

.(5)

The constant d_k are chosen to give u_k some predetermined scaling.

We will now formulate the analogous generalized conjugate-direction method for the minimization of a function $f(x)$.

Suppose that *U* and *V* form a pair conjugate set.

x_0 = arbitrary, $g_0 = g(x_0)$

For *i*= 0,1,..., compute

$x_{i+1} = x_i + \lambda_i u_i$, (6a)

Where λ_i minimizes $f(x_i + \lambda_i u_i)$ as a function of λ_i,

$\beta_i = -\lambda_i [v_i^T g_i / u_i^T (g_{i+1} - g_i)]$

(6b)

$x_{i+1} = x_i + \beta_i u_i$

(see VanWyk, 1977).

4. Standard pair Conjugate Algorithm:

Step (1): Set i=1.

Step (2): Compute , u_i=-g_i line search along d_i to get $x_{i+1} = x_i + \beta_i u_i$.

Step (3): If at x_{i+1} the stopping criterion $\| g_{i+1} \| \le 1 \times 10^{-5}$ is satisfied, then terminate.

Step (4): Check for restarting criterion if *i=n* then go to step (1). Else go to step (5).

Step (5): Compute $u_{i+1} = -g_{i+1} + \beta_i u_i$ where $\beta_i == -\lambda_i \left[\frac{v_i^T g_i}{y_i^T (g_{i+1} - g_i)} \right]$

Step(6): Set $i=i+1$.
Step (7): If $i > 1000$, stop. Else go to step 2.
(see VanWyk, 1977).

5. A New adaptive Minimization Procedure:

In this section we present a new adaptive optimal step size minimization scheme designed to improve the efficiency of the conjugate pair method. The technique gives only the trail step-size will be tested at each search, and needs only one additional gradient evaluation at each search. In eq.(3), the optimal step-size, λ_i is given by:

$$\lambda_i = -\frac{g_i^T g_i}{g_i^T G d_i}$$

Provided $g_i^T G d_i > 0$. However it requires knowledge of the Hessian G for a quadratic, which is undesirable. For more details see (Vrahatis et al (1996).

Therefore, we derive a formula without the need for the Hessian or its estimate.

We note that with a small value of $\varepsilon_i > 0$, we can define for a quadratic, the vector $f(x_i + \varepsilon_i d_i) - f(x_i) = g(x_i + \varepsilon_i d_i) - g(x_i) = \varepsilon_i G d_i$

Now, we can avoid computation of second derivatives by replacing the vector Gd_i in eq.(3) from the above, i.e. substituting

$$Gd_i = \frac{g(x_i + \varepsilon_i d_i) - g(x_i)}{\varepsilon_i} \qquad (7)$$

we get

$$\lambda_i = \frac{\varepsilon_i g_i^T g_i}{g_i^T (g(x_i + \varepsilon_i d_i) - g(x_i))^T} \qquad (8)$$

We shall use this formula to compute an optimal step-size estimation when the function f is not necessarily quadratic. We can now present the complete algorithm, which includes determining the step-size within the pair conjugate method.

6. New proposed algorithm (New1)

6.1 Outline of the proposed pair Conjugate Algorithm:

Step (0): Set i=0, select an initial point $x_i, \varepsilon_i = 1.e^{-1}, \phi = 0.5$, and compute g_i, set $u_i = -g_i, \varepsilon_i = \varepsilon$.

Step (1): Set i=i+1, for i<1, check that the ε_i is a suitable value, satisfying: $||\varepsilon_i d_i|| \leq \phi ||g_i||$, then compute $x_i = x_i + \varepsilon_i \beta_i u_i$, and go to step (2) else, set $\varepsilon_i = \varepsilon / 2$, and re-check.

Step (2): Compute the optimal step-size:

$$\lambda_i = -\frac{\varepsilon_i g_i^T g_i}{g_i^T(\overline{g_i} - g_i)}$$

where

$$\overline{g_i} = g(\overline{x_i})$$

Step (3): compute the new estimate minimum point:

$x_i^* = x_i + \hat{\lambda}_i u_i$ where λ_i^is the minimization of the function f (found by using cubic fitting technique)fully described in Bunday (1984).

Step (4): Use the following algorithm to find the new search direction:

$$u_{i+1} = -g_{i+1}^* + \beta_i u_i$$

where $\beta_i = \lambda_i \left(\frac{v_i^T g_i^*}{y_i^T y_i} \right)$

Step (5): Check that the new direction satisfies the descent property as $u_i^T g_i^* < 0$, and if so, go to step (6); otherwise go to step (0).

Step (6): check if $\| g^* \| < 1x10^{-5}$, Stop; otherwise, set $x_i = x_i^*$ and go to step (1).

7. An alternative new procedure for calculating the minimizer λi in step(3) of the above new proposed algorithm.

Armij rule Armijo(1966) is as follows

Give βЄ (0,1) ,ρ Є (0,0.5), τ>0 there exists the least non-negative integer m_k such that

$$f(x_k) - f(x_k + \beta^n \tau\, d_k) \geq -\rho \beta^m \tau\, g_k^T d_k \qquad \textbf{(9)}$$

In order to gurantee the objective function decreases sufficiently we will introduce a parameter α such that

$$f(x_k + \alpha_k d_k) \leq f(x_k) + \rho \alpha_k g_k^T d_k \qquad (10)$$

and

$$f(x_k + \alpha_k d_k) \geq f(x_k) + (1-\rho)\, \alpha_k g_k^T d_k \qquad (11)$$

where 0<ρ<0.5

Armijo proved that the above algorithm has superlinear conjugate , where

$$\alpha_k = \max\{\rho^{-j}, \qquad j = 0,1,2,.......\} \qquad (12)$$

Equation (10) is called Armijo liner search procedure. This equation can be modified further by the following steps.

Aimijo

$$f(x_k+\alpha_k d_k)-f(x_k)\le \rho\alpha_k g_k^T d_k \qquad (13)$$

Modified Armijo(1)

$$f(x_k+\alpha_k d_k)-f(x_k)\le \rho\alpha_k g_k^T d_k-\rho_1^2\alpha_k\|d_k\|^2 \qquad (14)$$

Modified Armijo(2)

$$f(x_k+\alpha_k d_k)-f(x_k)\le \rho\alpha_k g_k^T d_k-\rho_2^2\alpha_k\|g_k\|^2 \qquad (15)$$

$\rho_1,\rho_2>0$ *constant*

Note:- step(3) of algorithm (6.1) may be written again when the minimizer λ^ can be found by adaptive line –search procedures using (13),(14) or (15).

8. Conclusions and Numerical Results:

Several standard test functions were minimized to compare the new algorithm with standard **Pair Conjugate** algorithm. The same line search was employed in each of algorithms, namely the cubic interpolation procedure. We tabulate for all the algorithms the number of functions evaluations (NOF), the number of iterations (NOI). Overall totals are also given for NOF and N01 with each algorithm.

Table (1) gives the comparison between the standard pair Conjugate algorithm and the new algorithm, this table indicate that the new algorithm is better than the standard pair Conjugate algorithm.

Namely; talking the standard pair CG-method as 100% NOI and NOF we will get 75% NOI and 75% NOF. This means that there are 25% improved in both NOI and NOF.

Table (1): Comparative Performance of the Two Algorithms for Group of Test Function

Test function	N	Standared pair conjugate algorithm NOI (NOF)	The proposed algorithm NOI (NOF)
Powell	4 100 1000	100 (218) 120 (300) 215 (452)	97 (199) 101 (200) 100 (350)
Wood	4 100	223 (340) 451 (502)	210 (323) 368 (430)
Sum	100	50 (104)	27 (72)
Dixon	100	73 (159)	55 (48)
Rosen	100 1000	53 (120) 60 (144)	29 (76) 48 (112)
Cubic	100 1000	40 (105) 58 (122)	17 (49) 28 (99)
Tri	100	70 (145)	59 (87)
Total		1513 (2711)	1139 (2045)

The New Method:-

In order to find the point which minimizes a given functions. In this paper we have proposed a new conditions, and we are compared them with the Armijo's condition with respect to the maximum objective function. The new proposed conditions are follows:-

1. $f\left(x_k - t_k \nabla f(x_k)\right) - \max_{0\le j\le m}\left\{f(x_{k-j})\right\} \le -\sigma_1 t_k \|g_k\|^2$ (19)
2. $f\left(x_k - t_k \nabla f(x_k)\right) - \max_{0\le j\le m}\left\{f(x_{k-j})\right\} \le \sigma_2 t_k s_k^T g_k$ (20)
3. $f\left(x_k - t_k \nabla f(x_k)\right) - \max_{0\le j\le m}\left\{f(x_{k-j})\right\} \le \sigma_3 t_k \|s_k\|^2$ (21)
4. $f\left(x_k - t_k \nabla f(x_k)\right) - \max_{0\le j\le m}\left\{f(x_{k-j})\right\} \le -\sigma_1 t_k \|g_k\|^2 + \sigma_2 t_k s_k^T g_k$ (22)
5. $f\left(x_k - t_k \nabla f(x_k)\right) - \max_{0\le j\le m}\left\{f(x_{k-j})\right\} \le -\sigma_1 t_k \|g_k\|^2 + \sigma_3 t_k \|s_k\|^2$ (23)
6. $f\left(x_k - t_k \nabla f(x_k)\right) - \max_{0\le j\le m}\left\{f(x_{k-j})\right\} \le \sigma_2 t_k s_k^T g_k + \sigma_3 t_k \|s_k\|^2$ (24)
7. $f\left(x_k - t_k \nabla f(x_k)\right) - \max_{0\le j\le m}\left\{f(x_{k-j})\right\} \le 1+2+3$ (25)

Numerical Applications

The algorithm described in section ( ) has been implemented using the new FORTRAN program was tasted a Pentium (III) with random problems of varying dimensions. Our experience is that the algorithm behaved predictably and reliably and the results were quite satisfactory. Som typical computational results are given below. For the following problems, the reported parameters are:-

- n dimensions,
- $x^{\bullet} = (x_1, x_2, \ldots, x_n)$ starting point,
- $x^* = (x^*_1, x^*_2, \ldots, x^*_n)$ approximate local minimum computed within an accuracy of $\varepsilon = 10^{-4}$,

NOI the total number of iteration
NOF the total number of function

In table (1) we compare the numerical results obtained for various starting points, by applying other methods (Armijo's method, New proposed1, New proposed2, New proposed3, 4, 5, N6) including the classic conic method, with the corresponding numerical results of the method presented in this paper.

This table indicates the classical starting point. We used for all methods an accuracy of $\varepsilon_1 = 10^{-8}$ and an initial step 10. We also used $\varepsilon_3 = \varepsilon_4 = 10^{-15}$.

For our method we set the size of the line search record to be M=n. Our new propositional are better than the old method.

Table (2): Comparative performance between the standard conic method and new proposed method

Test function	N	Arm	New proposed 1	New proposed 2	New proposed 3
Rosen	20 80	30 (118) 49 (200)	28 (115) 49 (195)	20 (115) 50 (190)	19 (108) 30 (180)
Powell	4 100	33 (208) 97 (410)	20 (155) 40 (101)	21 (155) 41 (105)	18 (145) 30 (101)
Wood	4 40	39 (100) 300 (257)	31 (73) 220 (200)	32 (75) 220 (200)	28 (70) 199 (180)
Wolf	4 20	20 (30) 40 (101)	14 (27) 30 (71)	12 (25) 30 (71)	9 (15) 20 (60)
Dixon	10 200	51 (97) 104 (332)	35 (81) 100 (310)	30 (71) 160 (99)	25 (61) 25 (30)
Cubic	20 60	111 (320) 160 (501)	111 (320) 160 (501)	112 (322) 155 (500)	90 (270) 100 (403)
Total		1034 (2674)	838 (2149)	823 (1928)	593 (1623)

9. Appendix:

1- Generalized Powell Function:

$$f=\sum_{i=1}^{n/4}\left[(x_{4i-3}-10x_{4i-2})^2+5(x_{4i-1}-x_{4i})^2+(x_{4i-2}-2x_{4i-1})^4+10(x_{4i-3}-x_{4i})^4\right],$$

$$x_0=(3,-1,0,1;\cdots)^T.$$

2- Generalized Wood Function:

$$f=\sum_{i=1}^{n/4}100\left[(x_{4i-2}-x_{4i-3}^2)^2\right]+(1-x_{4i-3})^2+9(x_{4i}-x_{4i-1}^2)^2+(1-x_{4i-1}^2)^2$$

$$+10.1\left[(x_{4i-2}-1)^2+(x_{4i}-1)^2\right]+19.8(x_{4i-2}-1)^2(x_{4i}-1),$$

$$x_0=(-3,-1,-3,-1;\cdots)^T.$$

3- Generalized Sum of Quadratics Function:

$$f=\sum_{i=1}^{n}(x_i-1)^4,$$

$$x_0=(2;\cdots)^T.$$

4- Generalized Dixon Function:

$$f=\sum_{i=1}^{n}\left[(1-x_i)^2+(1-x_n)^2+\sum_{i=1}^{n-1}(x_i^2-x_{i-1})^2\right],$$

$$x_0=(-1;\cdots)^T.$$

5- Generalized Rosenbrock Function:

$$f = \sum_{i=1}^{n/2}\left[100\left(x_{2i} - x_{2i-1}^{2}\right)^{2} + \left(1 - x_{2i-1}\right)^{2}\right],$$

$$x_0 = \left(-1,2,1;\cdots\right)^T.$$

6- Generalized Cubic Function:

$$f = \sum_{i=1}^{n/2}\left[100\left(x_{2i} - x_{2i-1}^{3}\right)^{2} + \left(1 - x_{2i-1}\right)^{2}\right],$$

$$x_0 = \left(-1,2,1;\cdots\right)^T.$$

7- Generalized Tri Function:

$$f = \sum_{i=1}^{n}\left(ix_i^2\right)^2 ,$$

$$x_0 = \left(-1;\cdots\right)^T.$$

10. Reference

1) Al-Baali, M. (1985) "Descent Property and Global Convergence of the Fletcher-Reeves Method with Inexact Line Search" IMA Journal ofNumerical Analysis, Vol. 5, pp. 121-124.

2) Armijo L, (1966) (Minimization of function having Lipschitz Continuous partial derivatives) Pacific J. Math. vol. 16 pp 1-3.

3) Bazaraa, M. S. (2000) "Nonlinear Programming", England, Universities Press, London.

4) Bunday, B. D., (1984), "Basic Optimization Methods", Edward Arnold Hedford Square, London.

5) Davidon, W. C. (1959) "Variable Metric Method for Minimization". AERC and D. Report A. N1-5990. Argonne II.

6) Dixon, L. C. W. (1972) "Non-linear Optimization", English Universities press, London.

7) Gill, P. X. and Murray, W. (1974) "Numerical method for Constrained Optimization", Academic Press Inc. London.

8) Raydan, M. (1997), "The Barzilia and Borwein gradient method for the large scale unconstrained minimization problem", SIAM J. Optim., 7, 26-33.

9) Scales, L. E. (1985) "Introduction to Nonlinear Optimization" Macmillan, London.

10) VanWyk, D. J. (1977) "Generalization of Conjugate Direction in the Optimization of Function". JOTA, Vol. 21, No.4.pp. 435-450

11) Vrahatis, M. N., Androulakis, G. S. and Mnoussakis, G. E. (1996), "A New Unconstrained Optimization Method for Imprecise Function and Gradient Values", J. Math. Anal. Appl., 197, 586-607.