Iraqi Journal of Statistical Science (11) 2007
P.P. [1-26]

Two-point Stepsize Gradient Algoriyhms for Unconstrained Optimization

Abbas Y. Al-Bayati* **Huda I. Ahmed****

ABSTRACT

In this paper we have investigated three algorithms. In the first algorithm we have derived a new optimal step size gradient algorithm which is preferable over the classical SD algorithm both in theory and in the real computation. In the second algorithm we have derived and implemented a new formula for the non-quadratic model with a new ρ_i. In the third algorithm we have tried to make a new hybrid algorithm between the above three different step sizes.

Our numerical results are promising in general by implementing ten non-linear different test functions with different dimensions.

*Prof. / Department of Mathematics/College of Computer Sciences and Mathematics/Mosul University

** Lecture/ Department of Mathematics/College of Computer Sciences and Mathematics/Mosul University

Received: 24/ 1 /2006 Accepted: 31/ 8 / 2006

المستخلص

تم التقصي في هذا البحث عن ثلاث خوارزميات جديدة:
الخوارزمية الاولى تم اشتقاق خوارزمية التدرج ذات الخطوة المثلى والمفضلة على خوارزمية (SD) القياسية نظريا وعمليا.
الخوارزمية الثانية تم اشتقاق صيغة جديدة لنموذج غير تربيعي لاحتساب معلمة جديدة (ρ_i) وأثبتت كفاءتها عمليا.
الخوارزمية الثالثة تم التهجين بين الخطوات الثلاثة المثلى السابقة الذكر، بشكل عام

اثبتت النتائج العددية كفاءتها باستخدام عشر دوال غير خطية وابعاد مختلفة.

1. Introduction

Concerned the unconstrained optimization problem

$$\text{minimize } f(x) \qquad x \in R^n \tag{1}$$

where f is smooth and its gradient $g(x) = \nabla f(x)$ is available. The gradient algorithm for solving (1) is an iterative algorithm of the form $x_{k+1} = x_k - \alpha_k g_{k-1}$ (2)

where $g_{k-1} = \nabla f(x_{k-1})$ and α_k is a stepsize.

In the classical steepest descent algorithm (Cauchy, 1847); the stepsize is optioned by carrying out an exact line search, namely,

$$\alpha_k = \arg \min_{\alpha} f(x_k - \alpha_k g_k) \tag{3}$$

However, despite the optimal property of (3), the steepest descent algorithm performs poorly, its convergence linearly and is badly affected by ill conditioning (Akaike, 1959) and (Forsythe, 1968). Barzilai and Borwein (Barzilai and Borwein, 1988) proposed two-point stepsize gradient algorithm by regarding $H_k = \alpha_k I$ as an approximation to the Hessian inverse of f at x_k and imposing some QN condition on H_k (I is the identity matrix). Denote $s_{k-1} = x_k - x_{k-1}$ and $y_{k-1} = g_k - g_{k-1}$. By minimizing $\left\| H^{-1} s_{k-1} - y_{k-1} \right\|$, then obtained the following choice for the stepsize

$$\alpha_{k1} = \frac{s_{k-1}^T s_{k-1}}{s_{k-1}^T y_{k-1}}. \qquad (4)$$

Tthe motivation for this choice is that it provides two-point approximation to the secant equation underlying quasi-Newten method (Birgin and Etushenko, 1998). Raydan (Raydan, 1993) proved that the two-point stepsize gradient algorithm (2) and (4) is globally convergent. For the nonquadratic case, (Raydan, 1997) incorporated a globalization scheme of the two-point stepsize gradient method using the technique of non-monotone line search.

Dai and Yuan (Dai & Yuan, 2000) proposed another stepsize which is derived from the quadratic model and from the angle of interpolation which is very preferable than first stepsize defined in (4).

In this paper we have discussed three new algorithms, first we have investigated a new stepsize using the non-quadratic rational model and interpolation condition for derivation neither with steepest conjugate gradient algorithm, second, we have proposed a new formula discussed.

Third we have made a hybrid algorithm for those stepsize as a new case and show that the numerical result of this algorithm is competitive and sometime preferable over several famous conjugate gradient algorithms, especially for large scale unconstrained optimization.

1.1 The stepsize α_{k1} with the quadratic model (algorithm 1):

We list below the outline of the algorithm I:

For an initial point x_1

Step (1): set k=1, $d_k = -g_k$

Step (2): set $x_{k+1} = x_k + \alpha_k d_k$, where α_k (k=1) is a scalar chosen in such a way that $f_{k+1} < f_k$, if $k \neq 1$ compute α_k as:

$$\alpha_{k1} = \frac{s_{k-1}^T s_{k-1}}{s_{k-1}^T y_{k-1}}; \text{ where } s_{k-1} = x_k - x_{k-1} \text{ and } y_{k-1} = g_k - g_{k-1}$$

Step (3): check for convergence, i.e. if $\|g_{k+1}\| < \in$, where $\in$ is small positive tolerance, stop; otherwise continue.

Step (4): compute the new search direction defined by:

$d_k = -g_k + \beta_k d_{k-1}$, where β_k is computed by the following formula

$$\beta_k = \frac{g_k^T y_{k-1}}{d_{k-1}^T y_{k-1}} .$$

Step (5): if k=n or if the restarting criterion $\|g_k^T g_{k-1}\| > 0.2\|g_k\|^2$ (Powell, 1977) is satisfied go to step (1), else, set k=k+1, and go to step (2).

2. Derivation of the stepsize α_{k2} for the quadratic model:

Before the derivation of the new algorithm, let us see the work of Dai and Yuan (Dai et al, 2000) for the quadratic form:
They suppose that α_{k1} be defined by (4) and proposed that $t_k = \alpha_{k1}^{-1}$ so that

$$t_k = \frac{s_{k-1}^T y_{k-1}}{s_{k-1}^T s_{k-1}} \tag{5}$$

From the quadratic model $q_k(x)$:

$$q_k(x) = f_k + f_k' \ (x_k - x) + \frac{1}{2} f_k''(x_k - x)^2 \tag{6}$$

$$f(x_k + v_k) = f_k + f_k' \ v_k + \frac{1}{2} v_k^T \ G_k \ v_k \tag{7}$$

where $v_k = (x_{k+1} - x_k)$, similarly $v_k = \theta s_{k\text{-}1}$,so $f(x_k + \theta s_{k\text{-}1})$ is an approximation to the quadratic model ($q_k(\theta)$) so we have

$$q_k(\theta) = f_k + g_k^T \ \theta s_{k-1} + \frac{1}{2} G_k \ \theta^2 (s_{k-1})^2 \tag{8}$$

from quasi-Newton condition

$$H_k y_{k-1} = s_{k-1} \tag{9}$$

we have $y_{k-1} = H_k^{-1} s_{k-1} \ \Rightarrow \ \ y_{k-1} = G_k s_{k-1}$

so $$G_k = \frac{s_{k-1}^T y_{k-1}}{s_{k-1}^T s_{k-1}} = t_k \tag{10}$$

substitute (10) in (8) to get

$$q_k(\theta) = f_k + g_k^T \theta s_{k-1} + \frac{1}{2}\bar{b}_k \theta^2 (s_{k-1})^2 \qquad (11)$$

For any $t \in R$, the above model satisfies interpolation conditions

$$q_k(0) = f_k \qquad (12)$$

It is easy to test that if t is given by (5), the quadratic model (11) satisfies the interpolation condition

$$\nabla q_k(-1) = g_{k-1}^T s_{k-1} \qquad (13)$$

if (13) is replaced with another interpolation condition

$$q_k(-1) = f_{k-1} \qquad (14)$$

so (11) is becomes:

$$f_{k-1} = f_k - g_k^T s_{k-1} + \frac{1}{2}\bar{b}_k (s_{k-1})^2 \qquad (15)$$

$$t_k = \frac{2(f_{k-1} - f_k + g_k^T s_{k-1})}{s_{k-1}^T s_{k-1}} \qquad (16)$$

since $t_k = \alpha_k^{-1}$

so we have

$$\alpha_{k2} = \frac{s_{k-1}^T s_{k-1}}{2(f_{k-1} - f_k + g_k^T s_{k-1})} \qquad (17)$$

2.1 The stepsize α_{k2} for algorithm II :

We list below the outlines of algorithm 3.2:

For an initial point x_1

Step (1): set k=1, $d_k = -g_k$

Step (2): set $x_{k+1} = x_k + \alpha_k d_k$, where α_k (k=1) is a scalar chosen in such a way that $f_{k+1} < f_k$, if $k \neq 1$ compute α_k by:

$$\alpha_{k2} = \frac{s_{k-1}^T s_{k-1}}{2(f_{k-1} - f_k + g_k^T s_{k-1})}.$$

Step (3): check for convergence, i.e. if $\|g_{k+1}\| < \in$, where $\in$ is small positive tolerance, stop; otherwise continue.

Step (4): compute the new search direction defined by:

$d_k = -g_k + \beta_k d_{k-1}$, where β_k is computed by the following formula

$$\beta_k = \frac{g_k^T y_{k-1}}{d_{k-1}^T y_{k-1}} .$$

Step (5): if k=n or if $\|g_k^T g_{k-1}\| > 0.2\|g_k\|^2$ is satisfied, go to step (1), else, set k=k+1, and go to step (2).

Now, by using the same procedure of (Dai etal, 2000) let us derive the new formula for the α_{k3} but for the non-quadratic model.

3. Derivation of the new stepsize α_{k3} for the non-quadratic model

Since $q(x)$ is quadratic function then a function f is defined as nonlinear scaling of $q(x)$ if the following conditions holds

$$f = F(q(x)), \qquad \frac{df}{dq} = F' > 0 \text{ and } q(x) > 0 \tag{18}$$

Many authors have proposed special model as follows:

1- $F(q(x)) = (q(x))^p \qquad p > 0$ (Fried, 1971) (19)

2- $F(q(x)) = \ln q(x)$ (Al-Assady, 1991) (20)

3- $F(q(x)) = \dfrac{\in_1 q(x)}{1 - \in_2 q(x)}$ (Al-Bayati, 1993) (21)

4- $F(q(x)) = (\log(\in q(x) - 1), \quad \in > 0$ (Al-Bayati, 1995) (22)

5- $F(q(x)) = \sinh(\in q(x))$ (Al-Assady and Al-Ta'ai, 2002a) (23)

6- $F(q(x)) = \sin(\in q(x))$ (Al-Assady and Al-Ta'ai, 2002b) (24)

Now let $f_k = F(q_k(x))$ so we have

$$f'_k = g_k = F'_k\, G_k\, (x_k - x^*) \qquad (25)$$

$$f'' = F'_k\, G_k + G_k^2 (x_k - x^*)^2 F''_k \qquad (26)$$

Now since $q_k(\theta) = f_k + g_k^T \theta s_{k-1} + \frac{1}{2} G_k \theta^2 (s_{k-1})^2$

So we have

$$\begin{aligned} f_k(\theta) &= F_k(q(x)) + F'_k(q(x))\, G_k (x_k - x^*)\, \theta\, s_{k-1} + \\ &+ \frac{1}{2}[F'_k(q(x))\, G_k + F''_k(q(x))\, G_k^2\, (x_k - x^*)^2]\, \theta^2\, (s_{k-1})^2 \end{aligned} \qquad (27)$$

Substitute for G_k by t_k; for $(x_k - x^*)$ by $\theta (s_{k-1})$

$$\begin{aligned} f_k(\theta) &= F_k(q(x)) + F'_k(q(x))\, t_k\, \theta^2\, (s_{k-1})^2 + \frac{1}{2} F'_k(q(x))\, t_k\, \theta^2\, (s_{k-1})^2 + \\ &\frac{1}{2} F''_k(q(x))\, t_k^2\, \theta^4\, (s_{k-1})^4 \end{aligned} \qquad \textbf{(28)}$$

$$\begin{aligned} f_k(-1) &= F_k(q(x)) + F'_k(q(x))\, \bar{b}_k\, (s_{k-1})^2 + \frac{1}{2} F'_k(q(x))\, \bar{b}_k\, (s_{k-1})^2 + \\ &\frac{1}{2} F''_k(q(x))\, \bar{b}_k^2\, (s_{k-1})^4 \end{aligned} \qquad (29)$$

If we replace the interpolation condition with

$$f_k(-1) = f_{k-1} = F_{k-1}(q(x)) \qquad (30)$$

then we have

$$F_{k-1}(q(x)) = F_k(q(x)) + \frac{3}{2}F'_k(q(x))t_k\ (s_{k-1})^2 + \frac{1}{2}F''_k(q(x))\ t_k^2\ (s_{k-1})^4 \quad (31)$$

$$F_{k-1}(q(x)) - F_k(q(x)) = \frac{3}{2}F'_k(q(x))\ t_k\ (s_{k-1})^2 + \frac{1}{2}F''_k(q(x))\ t_k^2\ (s_{k-1})^4 \quad (32)$$

$$t_k^2(\frac{1}{2}F''_k(q(x))\,(s_{k-1})^4) + t_k(\frac{3}{2}F'_k(q(x))\,(s_{k-1})^2) - [F_{k-1}(q(x)) - F_k(q(x))] = 0 \quad (33)$$

$$t_k = \frac{-\frac{3}{2}F'_k(q(x))\,(s_{k-1})^2 \pm \sqrt{(\frac{3}{2}F'_k(q(x))\,(s_{k-1})^2)^2 + 4[\frac{1}{2}F''_k(q(x))\,(s_{k-1})^4\,(F_{k-1}(q(x)) - F_k(q(x)))]}}{2(\frac{1}{2}F''_k(q(x))\,(s_{k-1})^4)} \quad (34)$$

since $t_k = \alpha_k^{-1}$

so we have

$$\alpha_{k3} = \frac{F''_k(q(x))\,(s_{k-1})^4}{-\frac{3}{2}F'_k(q(x))\,(s_{k-1})^2 \pm \sqrt{(\frac{3}{2}F'_k(q(x))\,(s_{k-1})^2)^2 + 2[F''_k(q(x))\,(s_{k-1})^4\,(F_{k-1}(q(x)) - F_k(q(x)))]}} \quad (35)$$

Now we are going to derive a new ρ_k for the proposed new exponential non-quadratic model.

3.1 A new non-quadratic model for unconstrained optimization

In this section a new exponential function (base a, a>0) model is investigated and tested on a set of standard nonlinear unconstrained test function, it is assumed that condition (18) holds. The new model is follows:

$$F(q(x)) = a^{\frac{\in_1 q(x)}{\in_2(1-q(x))}} \qquad ,\in_1,\in_2 > 0 \quad (36)$$

$$F(q(x)) = e^{\frac{\in_1 q(x)}{\in_2 (1-q(x))} \ln a} \quad (37)$$

$$\ln F = \ln(e^{\frac{\in_1 q(x)}{\in_2 (1-q(x))} \ln a}) \quad (38)$$

$$\ln F = \frac{\in_1 q(x)}{\in_2 (1-q(x))} \ln a \quad (39)$$

and we first assume that neither $\in_1$ nor $\in_2$ is zero in (39),solving (39) for $q(x)$ gives

$$\in_2 \ln F - \in_2 q(x) \ln F = \in_1 q(x) \ln a \quad (40)$$

$$\in_2 \ln F = \in_1 q(x) \ln a + \in_2 q(x) \ln F$$

$$\in_2 \ln F = (\in_1 \ln a + \in_2 \ln F)\, q(x)$$

$$q = \frac{\in_2 \ln F}{\in_1 \ln a + \in_2 \ln F} \quad (41)$$

$$F' = a^{\frac{\in_1 q}{\in_2 (1-q)}} \frac{\in_1 \in_2 (1-q) + \in_1 \in_2 q}{(\in_2 (1-q))^2} \ln a \quad (42)$$

$$F' = a^{\frac{\in_1 q}{\in_2 (1-q)}} \frac{\in_1 \in_2}{(\in_2 (1-q))^2} \ln a \quad (43)$$

F' in (43) contains three parts

(i) $a^{\frac{\in_1 q}{\in_2 (1-q)}}$, (ii) $\frac{\in_1 \in_2}{(\in_2 (1-q))^2}$, (iii) $\ln a$

first we find the part (i) and substitute (41) in this equation

$$\frac{\in_1 q}{\in_2 (1-q)} = \frac{\in_1 \left(\frac{\in_2 \ln F}{\in_1 \ln a + \in_2 \ln F}\right)}{\in_2 \left(1 - \frac{\in_2 \ln F}{\in_1 \ln a + \in_2 \ln F}\right)} = \frac{\in_1 \in_2 \ln F}{\in_1 \ln a + \in_2 \ln F} \; \frac{\in_1 \ln a + \in_2 \ln F}{\in_1 \in_2 \ln a} = \frac{\ln F}{\ln a}$$

then $a^{\frac{\in_1 q}{\in_2 (1-q)}} = a^{\frac{\ln F}{\ln a}} = e^{\frac{\ln F}{\ln a} \ln a} = e^{\ln F} = F$ (44)

Now to find the value of (ii)

$$\frac{\in_1 \in_2}{(\in_2 (1-q))^2} = \frac{\in_1 \in_2}{\left[\in_2 \left(1 - \frac{\in_2 \ln F}{\in_1 \ln a + \in_2 \ln F}\right)\right]^2} = \frac{\in_1 \in_2}{\left[\in_2 \left(\frac{\in_1 \ln a}{\in_1 \ln a + \in_2 \ln F}\right)\right]^2}$$

$$= \frac{\in_1}{\in_2 \left(\frac{\in_1 \ln a}{\in_1 \ln a + \in_2 \ln F}\right)^2} = \frac{\in_1}{\in_2} \frac{(\in_1 \ln a + \in_2 \ln F)^2}{\in_1^2 \ln a^2} = \frac{(\in_1 \ln a + \in_2 \ln F)^2}{\in_1 \in_2 \ln a^2}$$

i.e. $\frac{\in_1 \in_2}{(\in_2 (1-q))^2} = \frac{(\in_1 \ln a + \in_2 \ln F)^2}{\in_1 \in_2 \ln a^2}$ (45)

Now substitute (44) and (45) in (43) then we get

$$F' = F \, \frac{(\in_1 \ln a + \in_2 \ln F)^2}{\in_1 \in_2 \ln a^2} \, \ln a \qquad (46)$$

$$F' = F \, \frac{(\in_1 \ln a + \in_2 \ln F)^2}{\in_1 \in_2 \ln a} = F \, \frac{(\in_2 (\frac{\in_1}{\in_2} \ln a + \ln F))^2}{\in_1 \in_2 \ln a} = F \frac{\in_2^2 (\frac{\in_1}{\in_2} \ln a + \ln F)^2}{\in_1 \in_2 \ln a}$$

then we get

$$F' = F \frac{\in_2}{\in_1 \ln a} (\frac{\in_1}{\in_2} \ln a + \ln F)^2 \qquad (47)$$

Now to find the value of ρ_k let

$$\rho_k = \frac{F'_{k-1} q_{k-1} + \omega}{F'_k q_k} \qquad (48)$$

where $\omega = \dfrac{\alpha_k g_{k-1}^T d_{k-1}}{2}$ (49)

since $\rho_k = \dfrac{F'_{k-1}}{F'_k}$ (50)

so $\rho_k = \rho_k \dfrac{q_{k-1}}{q_k} + \dfrac{\omega}{F'_k q_k}$ (51)

multiply both sides of (51) by q_k

$$\rho_k q_k = \rho_k q_{k-1} + \frac{\omega}{F'_k} \quad (52)$$

$$\rho_k (q_k - q_{k-1}) = \frac{\omega}{F'_k} \quad (53)$$

to find the value of (53) we first find the value of $(q_k - q_{k-1})$ we use (41), then we have

$$q_k - q_{k-1} = \frac{\in_2 \ln F_k}{\in_1 \ln a + \in_2 \ln F_k} - \frac{\in_2 \ln F_{k-1}}{\in_1 \ln a + \in_2 \ln F_{k-1}}$$

$$= \frac{\in_1 \in_2 \ln a (\ln F_k - \ln F_{k-1})}{(\in_1 \ln a + \in_2 \ln F_k)(\in_1 \ln a + \in_2 \ln F_{k-1})} \quad (54)$$

substitute $\boldsymbol{\rho_k}$ defined by (49) in (53) then we get

$$\frac{F'_{k-1}}{F'_k}(q_k - q_{k-1}) = \frac{\omega}{F'_k} \quad (55)$$

multiply both sides by F'_k

$$F'_{k-1}(q_k - q_{k-1}) = \omega \quad (56)$$

using (47) and (54) in (56)to get

$$F_{k-1}\frac{\in_2}{\in_1 \ln a}\left(\frac{\in_1}{\in_2}\ln a+\ln F_{k-1}\right)^2\frac{\in_1\in_2 \ln a(\ln F_k-\ln F_{k-1})}{\in_2^2\left(\frac{\in_1}{\in_2}\ln a+\ln F_k\right)\left(\frac{\in_1}{\in_2}\ln a+\ln F_{k-1}\right)}=\omega$$

$$\frac{(\ln F_k-\ln F_{k-1})\left(\frac{\in_1}{\in_2}\ln a+\ln F_{k-1}\right)F_{k-1}}{\left(\frac{\in_1}{\in_2}\ln a+\ln F_k\right)}=\omega \qquad (57)$$

now let

$$(\ln F_k-\ln F_{k-1})=\xi \qquad (58)$$

$$\left(\frac{\in_1}{\in_2}\ln a+\ln F_{k-1}\right)=\psi \qquad (59)$$

and $\left(\frac{\in_1}{\in_2}\ln a+\ln F_k\right)=\frac{\in_1}{\in_2}\ln a+\ln F_{k-1}-\ln F_{k-1}+\ln F_k=\psi+\xi$ (60)

substitute (58), (59) and (60) in (57)

$$\frac{\xi\,\psi\,F_{k-1}}{\psi+\xi}=\omega \qquad (61)$$

$$\xi\psi F_{k-1}=\omega\psi+\omega\xi$$

$$\psi(\xi F_{k-1}-\omega)=\omega\xi$$

$$\psi=\frac{\xi\omega}{\xi F_{k-1}-\omega} \qquad (62)$$

add $(-\ln F_{k-1})$ for the both sides of (62), then

$$\psi-\ln F_{k-1}=\frac{\xi\omega}{\xi F_{k-1}-\omega}-\ln F_{k-1}$$

since $\left(\frac{\in_1}{\in_2}\ln a+\ln F_{k-1}\right)=\psi$ which defined in (59) so we have

$$\frac{\in_1}{\in_2}\ln a = \frac{\xi\omega}{\xi F_{k-1} - \omega} - \ln F_{k-1} \tag{63}$$

Now to find the value of ρ_k, let

$$\rho_k = \frac{F'_{k-1}}{F'_k} = \frac{F_{k-1}\dfrac{\in_2}{\in_1 \ln a}(\dfrac{\in_1}{\in_2}\ln a + \ln F_{k-1})^2}{F_k\dfrac{\in_2}{\in_1 \ln a}(\dfrac{\in_1}{\in_2}\ln a + \ln F_k)^2}$$

so $$\rho_k = \frac{F_{k-1}(\dfrac{\in_1}{\in_2}\ln a + \ln F_{k-1})^2}{F_k(\dfrac{\in_1}{\in_2}\ln a + \ln F_k)^2} \tag{64.a}$$

For simplicity (3.62a) may be simplified to

$$\rho_k = \frac{F_{k-1}\psi^2}{F_k(\psi+\zeta)^2} = \frac{F_{k-1}\left(\dfrac{\xi\omega}{\xi F_{k-1} - \omega}\right)^2}{F_k\left(\dfrac{\xi\omega}{\xi F_{k-1} - \omega} + \xi\right)^2} = \frac{1}{F_k F_{k-1}}\left(\frac{\omega}{\xi}\right)^2 \tag{64.b}$$

to compute ρ_k we may use either (64.a) or (64.b).

3.2 The optimal stepsize α_{k3} with the non-quadratic model (algorithm III):

We list below the outlines of the new algorithm III

For an initial point x_1

Step (1): set k=1, $d_k = -g_k$

Step (2): set $x_{k+1} = x_k + \alpha_k d_k$, where α_k in (k=1) is a scalar chosen in such a way that $f_{k+1} < f_k$, if $k \neq 1$ compute α_k by

$$\alpha_{k3} = \frac{F_k''(q(x))\,(s_{k-1})^4}{-\frac{3}{2}F_k'(q(x))\,(s_{k-1})^2 \pm \sqrt{(\frac{3}{2}F_k'(q(x))\,(s_{k-1})^2)^2 + 2[F_k''(q(x))\,(s_{k-1})^4(F_{k-1}(q(x)) - F_k(q(x)))]}}$$

Step (3): check for convergence, i.e. if $\|g_{k+1}\| < \in$, where $\in$ is small positive tolerance, stop; otherwise continue.

Step (4): compute $con = \frac{\xi\omega}{\xi F_{k-1} - \omega} - \ln F_{k-1}$,

where $\xi = \ln F_k - \ln F_{k-1}$ and $\omega = \frac{\alpha_k g_{k-1}^T d_{k-1}}{2}$.

Step (5): Compute the new $\rho_k = \frac{F_{k-1}(con + \ln F_{k-1})^2}{F_k(con + \ln F_k)^2}$.

Step (6): Compute the new search direction defined by:
$d_k = -g_k + \rho_k \beta_k d_{k-1}$, where β_k is computed by the following formula $\beta_k = \frac{g_k^T y_{k-1}}{d_{k-1}^T y_{k-1}}$.

Step (7): if k=n or if $\|g_k^T g_{k-1}\| > 0.2\|g_k\|^2$ is satisfied go to step (1), else, set k=k+1, and go to step (2).

4. A new hybrid algorithm for the three optimal stepsizes

We have discussed earlier in this chapter three different stepsizes for both quadratic and non-quadratic models. In this section it is important to make an interleaving scheme between those stepsizes mentioned in (3.2), (17) and (35) respectively. All the details may be found in the following new algorithm.

4.1 New hybrid algorithm (algorithm IV):

We list below the outlines of the new procedure:

For an initial point x_1

Step (1): set k=1, $d_k = -g_k$.

Step (2): compute α_k, where α_k in (k=1) is a scalar chosen in such a way that $f_{k+1} < f_k$, if $k \neq 1$ compute α_k by

$$\alpha_{k3} = \frac{F_k''(q(x))\,(s_{k-1})^4}{-\frac{3}{2}F_k'(q(x))\,(s_{k-1})^2 \pm \sqrt{(\frac{3}{2}F_k'(q(x))\,(s_{k-1})^2)^2 + 2[F_k''(q(x))\,(s_{k-1})^4\,(F_{k-1}(q(x)) - F_k(q(x)))]}}$$

Step (3): if $0 < \alpha_{k3} < 1$, set $\alpha_k = \alpha_{k3}$ and go to step (5), else compute $\alpha_{k2} = \dfrac{s_{k-1}^T s_{k-1}}{2(f_{k-1} - f_k + g_k^T s_{k-1})}$.

Step (4): if $0 < \alpha_{k2} < 1$, set $\alpha_k = \alpha_{k2}$ and go to step (5), else compute $\alpha_{k1} = \dfrac{s_{k-1}^T s_{k-1}}{s_{k-1}^T y_{k-1}}$.

Step (5): set $x_{k+1} = x_k + \alpha_k d_k$.

Step (6): check for convergence, i.e. if $\|g_{k+1}\| < \in$, where $\in$ is small positive tolerance, stop; otherwise continue.

Step (7): compute $con = \dfrac{\xi\omega}{\xi F_{k-1} - \omega} - \ln F_{k-1}$,

where $\xi = \ln F_k - \ln F_{k-1}$ and $\omega = \dfrac{\alpha_k g_{k-1}^T d_{k-1}}{2}$.

Step (8): Compute $\rho_k = \dfrac{F_{k-1}(con + \ln F_{k-1})^2}{F_k(con + \ln F_k)^2}$

Step (9): Compute β_k; the conjugate coefficient, which is defined as:

β_{kq} = Is conjugate coefficient of quadratic model defined by

$$\beta_{kq} = \frac{g_k^T y_{k-1}}{d_{k-1}^T y_{k-1}}.$$

β_{kn} = Is conjugate coefficient of non-quadratic model defined by

$$\beta_{kn} = \rho_k \frac{g_k^T y_{k-1}}{d_{k-1}^T y_{k-1}}.$$

Step (10): check if $0 < \beta_k < 1$, set $\beta_k = \beta_{kq}$ and go to step (11), else set $\beta_k = \beta_{kn}$, continue.

Step (11): compute the new search direction defined by:

$$d_k = -g_k + \beta_k d_{k-1}.$$

Step (12): if k=n or if $\|g_k^T g_{k-1}\| > 0.2\|g_k\|^2$ is satisfied, go to step (1), else, set k=k+1, and go to step (2).

5. Numerical results and conclusions

We have tested four algorithms with double precisions, the first two algorithms are based on stepsizes α_{k1}, α_{k2} defined on quadratic model and the third new algorithm is based on α_{k3} for non-quadratic model, the fourth new algorithm is the new hybrid among the stepsizes (α_{k1}, α_{k2}, α_{k3}).

The stopping condition is $\|g_k\| < 10^{-5}$, a cubic fitting procedure which was described in details by Bundy (Bundy, 1984) used as a line search procedure.

All the results are obtained by using Pentium 4 and all programs are written in FORTRAN language. The comparative performance for all of these algorithms are evaluated by considering NOF, NOI.

We compared these stepsizs α_{k1}, α_{k2}, α_{k3} with the hybrid algorithm. Our numerical results are reported in Table (3.1) with dimension n=50, Table (1) with n=100, Table (2) with n=1000, Table (3) with n=10000, with different non-linear unconstrained test functions (see Appendix).

The comparative performance for all of these algorithms are evaluated by considering NOF, NOI where NOF is the number of function evaluations, NOI is the number of iterations.

The gains are some time significant for example, Powell with n=50, 100 and Cantrel with n=1000 and Rosen with all dimensions. Therefore our numerical results suggest two efficient modified stepsize:

1- stepsize gradient algorithm α_{k3} with non-quadratic model.

2- the hybrid algorithm with the stepsize $\alpha_{k1},\alpha_{k2},\alpha_{k3}$.

Which require few more storages and computational effectiveness in every iteration.

Table (1):Numerical comparisons of the new gradient algorithm and hybrid algorithm (n=50)

Function	α_{k1}		α_{k2}		α_{k3}		hybrid	
	NOF	NOI	NOF	NOI	NOF	NOI	NOF	NOI
Powell	124	35	151	38	112	31	108	31
Wood	84	26	81	27	87	26	96	29
Canterl	176	29	169	33	101	25	132	19
Rosen	161	42	200	32	92	30	92	30
Cubic	64	15	56	13	44	13	44	13
Sum	75	12	80	12	78	12	78	12
Edeger	15	5	21	5	18	5	16	5
Beal	34	10	29	9	39	9	34	11
Shallow	26	9	24	8	30	8	24	8
Wolfe	141	47	143	47	141	47	141	47
Total	900	240	981	244	742	206	735	199

Table (2)

Numerical comparisons of the new gradient algorithm and hybrid algorithm (n=100)

Function	α_{k1}		α_{k2}		α_{k3}		hybrid	
	NOF	**NOI**	NOF	NOI	NOF	NOI	NOF	NOI
Powell	124	35	151	38	158	37	114	30
Wood	184	39	181	38	112	30	114	37
Canterl	276	33	296	33	292	37	137	21
Rosen	163	43	695	174	97	30	97	30
Cubic	64	15	56	13	44	13	44	13
Sum	88	14	80	13	99	14	91	14
Edeger	15	5	21	5	18	5	16	5
Beal	34	10	34	10	46	11	29	9
Shallow	26	9	24	8	36	9	24	8
Wolfe	150	50	153	49	147	49	141	49
Total	1124	235	1691	381	1049	235	796	206

Table (3)

Numerical comparisons of the new gradient algorithm and hybrid algorithm (n=1000)

Function	α_{k1}		α_{k2}		α_{k3}		hybrid	
	NOF	**NOI**	NOF	NOI	NOF	NOI	**NOF**	**NOI**
Powell	163	42	151	38	138	45	151	46
Wood	187	37	181	37	104	27	89	27
Canterl	276	22	258	27	162	22	101	17
Rosen	163	43	696	174	103	29	103	29
Cubic	66	16	56	13	44	13	44	13
Sum	155	30	165	29	220	33	147	24
Edeger	17	6	23	6	23	6	18	6
Beal	34	10	34	10	41	9	30	9
Shallow	26	9	24	8	33	8	28	9
Wolfe	182	61	163	52	148	50	148	50
Total	1269	276	1751	394	1016	242	829	230

Table (4)

Numerical comparisons of the new gradient algorithm and hybrid algorithm (n=10000)

Function	α_{k1}		α_{k2}		α_{k3}		hybrid	
	NOF	**NOI**	NOF	NOI	NOF	NOI	NOF	NOI
Powell	173	44	172	41	245	57	184	45
Wood	**87**	**27**	81	27	122	29	95	31
Canterl	253	27	258	27	291	31	230	26
Rosen	163	43	696	174	113	27	110	32
Cubic	66	16	58	14	44	13	44	13
Sum	184	34	222	34	377	37	180	32
Edeger	17	6	23	6	27	6	18	6
Beal	34	10	34	10	52	10	34	10
Shallow	26	9	28	9	40	9	28	9
Wolfe	523	179	531	170	529	170	517	165
Total	1726	415	2303	532	1640	382	1440	369

Appendix:

These test functions are famous and form general literature

1- Generalized Powell function:

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

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

2- Generalized Wood function:

$$f(x)=\sum_{i=1}^{n/4}100[(x_{4i-2}-x_{4i-3}^2)^2]+(1-x_{4i-3})^2+90(x_{4i}-x_{4i-1}^2)^2+(1-x_{4i-1})^2$$

$$+10.1((x_{4i-2}-1)^2+(x_{4i}-1)^2)+19.8(x_{4i-2}-1)(x_{4i}-1),$$

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

3- Generalized Cantrel function:

$$f(x)=\sum_{i=1}^{n/4}[(\exp(x_{4i-3})-x_{4i-2})^4+100(x_{4i-2}-x_{4i-1})6+\arctan(x_{4i-1}-x_{4i})^4+x_{4i-3},$$

$$x_0=(1,2,2,2;...)^T.$$

4- Generalized Rosenbrock function:

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

$$x_0=(-1.2,1;...)^T.$$

5- Generalized Cubic function:

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

$$x_0 = (-1.2,1;...)^T.$$

6- Generalized Sum function:

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

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

7- Generalized Edeger function:

$$f(x) = \sum_{i=1}^{n} [(x_{2i-1} - 2)^4 + (x_{2i-1} - 2)^2 * x_{2i}{}^2 + (x_{2i} + 1)^2],$$

$$x_0 = (1,0;...)^T.$$

8- Generalized Beal function:

$$f(x) = \sum_{i=1}^{n} [(1.5 - x_{2i-1}(1 - x_{2i}^2)) + (2.25 - x_{2i-1}(1 - x_{2i}^2))^2 + [2.625 - x_{2i-1}(1 - x_{2i}^3))^2],$$

$$x_0 = (0,0;...)^T.$$

9- Generalized Shallow function:

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

$$x_0 = (-2;...)^T.$$

10- Generalized Wolfe function:

$$f(x) = [-x_1(3 - x_1/2 + 2x_2 - 1]^2 + \sum_{i=1}^{n-1} [(x_{i-1} - x_i(3 - x_i/2) + 2x_{i+1}]^2$$

$$+ [x_{n-1} - x_n(3 - x_n/2) - 1]^2,$$

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

References

Akaike, H., (1959), "On a successive transformation of probability distribution and its application to the analysis of the optimum gradient method, Ann. Inst. Statis. Math. Tokyo, Vol. 11, pp. 1-17.

Al-Assady, N. H., (1991),"New QN and PCG Algorithms Based on Non-Quadratic Properties", J. Ed and Sc., 12.

Al-Assady, N. H. and Al-Ta'ai, B. A., (2002a)," A New Non-Quadratic Model for Unconstrained Non-Linear Optimization" University of Computer and Mathematical Science, No. 48, 2002.

Al-Assady, N. H. and Al-Ta'ai, B. A., (2002b)," Hybrid Model for Unconstrained Non-Linear Optimization" University of Computer and Mathematical Science, No. 48, 2002.

Al-Bayati, A. Y., (1993),"A new Non-Quadratic Model for Unconstrained Non-linear Optimization Method" Natural and applied series Mu'tah Journal for Research and Studies Mu'tah University, Jordan, Vol. 8,No. 6, pp. 133-155.

Al-Bayati, A. Y., (1995),"A New Extended CG-method for Non-linear Optimization Method" Natural and applied series Mu'tah Journal for Research and Studies Mu'tah University, Jordan, Vol 10,No. 6, pp. 69-87.

Al-Bayati, A.Y. and Ahmed, I. H., (2005a), "A new four-parameter family of nonlinear conjugate gradient methods", Iraq Journal of Statistical Science, University of Mosul, College of Mathematics and Computer Science, Vol. 7, pp. 18-38.

Al-Bayati, A.Y. and Ahmed, I. H., (2005b), "Global convergence of the new four-parameter family of CG-methods without line search", First Arabic conference for Mathematics, Applied Science University (private), Jordan, forthcoming.

Barzilai, J. and Borwein, M., (1988), "Two point step size gradient methods", IMA, Journal of Numerical Analysis, Vol. 8, pp.141-148.

Bundy, B., (1984),"Basic Optimization Method", Edward Arnold, Bedford Square,London, U. K.

Dai, Y. H. ,Yaun, Y. And Yaun, J., (2000), "Modified Two-Point Stepsize Gradient Methods For Unconstrained Optimization", Numerical Linear Algebra and Optimization, Chinese NSF grants.

Forsythe, E. G. (1968), "On the asymptotic directions of the s-dimensional optimum gradient method, Numerische Mathematik, Vol. 11, pp. 57-76.

Powell, M. J. D., (1977), "Restart Procedures for the Conjugate Gradient Method", Mathematical Programming, Vol. 12, pp. 241-254.

Raydan, M., (1993), "On the Barzilai and Borwein choice of steplength for the gradient method", IMA ,Journal of Numerical Analysis, Vol. 13, pp. 321-326.

Raydan, M., (1997),"On the Barzilai and Borwein gradient method for the large scale unconstrained minimization problem", SAIM J. Optim. , 7:1, pp. 26-33.