# A New hybrid generalized CG- method for non-linear functions

**Abbas Y. Al-Bayati** **Hamsa Th. Chilmerane**
*profabbasalbayati@yahoo.com* *hamsathrot@uomosul.edu.iq*
*College of Computer Sciences and Mathematics*
*University of Mosul*

**Received on: 6/4/2005** **Accepted on: 9/10/2005**

## ABSTRACT

In this paper a new extended generalized conjugate gradient algorithm is proposed for unconstrained optimization, which is considered as anew inverse hyperbolic model .In order to improve the rate of convergence of the new technique, a new hybrid technique between the standard F/R CG-method and Sloboda CG-method using quadratic and non-quadratic models is proposed by using exact and inexact line searches. This method is more efficient and robust when applied on number of well-known nonlinear test function.

**Keywords:** F/R, exact line search, inexact line search.

## خوارزمية تهجين جديدة لطريقة التدرج المترافق للدوال غير الخطية

**عباس يونس البياتي** **همسة ثروت جلميران**
*كلية علوم الحاسوب والرياضيات*
*جامعة الموصل*

**تاريخ استلام البحث: 2005/4/6** **تاريخ قبول البحث: 2005/10/9**

## الملخص

في هذا البحث تم استخدام توسيع جديد من خوارزميات التدرج المترافق المعمم في الامثلية غير المقيدة باستخدام تقنيه جديدة مركزه على فكره التهجين .من اجل تحسين التقارب الشامل لطريقه التدرج المترافق المشروط استخدمنا فكره التهجين بين خوارزميتين للمتجهات المترافقة وهما خوارزميتا Sloboda,F/Rللتدرج المترافق المشروط باستعمال نماذج تربيعه وغير ألتربيعه وباستخدام خطوط بحث تامة وغير تامة اختبرت هذه الخوارزمية وذلك بتطبيقها على العديد من المسائل غير الخطية غير المقيدة المعروفة وقد أثبتت كفاءتها مقارنه مع مثيلاتها من خوارزميات للتدرج المترافق لدوال غير خطية.

**الكلمات المفتاحية:** F/R، خط بحث تام، خط بحث غير تام.

## 1. Introduction

The conjugate gradient method is particularly useful for minimizing function of mult variables because it does not require the storage of any matrices. However, the rate of convergence of the algorithm is only linear unless the iterative procedure is "restarted" occasionally .At present it is usual to restart every *n* or *(n+1)* iterations, where *n* is the number of variables, but it is known that frequency of restarts should depend on the objective function.

$$d_{k+1} = \begin{bmatrix} -g_k & for \quad k=1 \\ -g_{k+1} + \beta_k d_k & fork \succ 1, \end{bmatrix}$$

CG method is useful to find minimum of a function $f : R^n \longrightarrow R$. In general, the method has the following form

$$x_{k+1} = x_k + \lambda_k d_k \qquad ...(1)$$

where $g_k$ denotes the gradient $\nabla f(x_k)$, $d_k$ is the search direction, $\lambda_k$ is a steplength obtained by a line search, and $\beta_k$ is chosen so that it becomes the $k$-th conjugate direction when the function is quadratic and the line search is exact ,a well known formula for $\beta_k$ is given by

$$\beta_k = \frac{y_k^T g_{k+1}}{d_k^T y_k} \quad \text{( Hestenes \& Stiefel , 1952 )} \qquad \ldots(2)$$

$$\beta_k = \frac{\|g_{k+1}\|^2}{\|g_k\|^2} \quad \text{(Fletcher \& Reeves, 1964)} \qquad \ldots(3)$$

$$\beta_k = \frac{g_{k+1}^T (g_{k+1} - g_k)}{\|g_k\|^2} \quad \text{( Polak \& Ribire , 1969 )} \qquad \ldots(4)$$

$$\beta_k = \frac{-\|g_{k+1}\|^2}{d_k^T g_k} \quad \text{( Dixon , 1975 )} \qquad \ldots(5)$$

$$\beta_k = \frac{-g_{k+1}^T y_k}{d_k^T g_k} \quad \text{( Al-Bayati \& Al-Assady , 1986 )} \qquad \ldots(6)$$

where $y_k = g_{k+1} - g_k$ …(7)

The successive directions are conjugate vectors for the successive gradients obtained as the method progresses.

**2. Non Quadratic Models:**

The conjugate gradient methods so far discussed is a local quadratic representation of the objective function .In problems when the quadratic representation is not good, or when we are remote from such a region, quadratic function *F(q(x))* ,where *F* is monotonic increasing ,may be better to represent the objective and thus it gives an advantage to algorithm based on this model. In order to obtain better global rate of convergence for minimization algorithms when applied to more general functions than the quadratic. In this paper several new algorithms, which are invariant to nonlinear scaling of quadratic functions are proposed. There is some precedent for this approach, if $q(x)$ is quadratic function then a function *f* is defined as nonlinear scaling of $q(x)$ if the following condition holds

$$f = F(q(x)) \qquad \frac{df}{dq} = F' \rangle 0 \qquad \text{and} \qquad q(x) \rangle 0$$

where $x^*$ is minima of *q(x)* with respect to *x*.

The following properties are immediately derived from the above conditions.

**i** - every contour line of *q(x)* is a contour line of *f* ;

**ii** - if *x** is minimizer of *q(x)* then it is a minimizer of *f ;*

**iii** - that *x** is global minimum of *q(x)* does not necessarily mean that it is a global minimum of *f*. For more details see Boland et al., (1979).

Many authors have proposed special models as follows:

$$F(q(x)) = (q(x))^{p} \qquad p \succ 0$$ Fried (1971) …(8)

$$F(q(x)) = \in_1 q(x) + \frac{1}{2} \in_2 q^2(x) \qquad \in_1, \in_2 \ scaler$$ Boland et al., (1979) …(9)

$$F(q(x)) = \frac{\in_1 q(x)}{1 - \in_2 q(x)} \qquad \in_2 \prec 0$$ Al-Bayati (1993) …(10)

$$F(q(x)) = (\log(\in q(x)) - 1 \ ) \qquad \in \succ 0$$ Al-Bayati (1995) …(11)

$$F(q(x)) = \log\left(\frac{\in_1 q(x)}{\in_2 q(x) + 1}\right) \qquad \in_2 \prec 0$$ Al-Assady and Huda (1997) …(12)

$$F(q(x)) = \sin(\in q(x))$$ Al-Assady and Al-Ta'ai (2002) …(13)

**3. New Extended Generalized Conjugate Gradient Method by using Hyperbolic Inverse Model:**

Let

$$f_k = \sinh^{-1} q_k \qquad \ldots(14)$$

Then

$$f_k = \ln(q_k + \sqrt{1+q_k^2}) \qquad \ldots(15)$$

$$e^{f_k} = q_k + \sqrt{1+q_k^2} \qquad \ldots(16)$$

$$e^{f_k} - q_k = \sqrt{1+q_k^2} \qquad \ldots(17)$$

$$e^{2f_k} - 2e^{f_k} q_k + q_k^2 = 1 + q_k^2 \qquad \ldots(18)$$

$$e^{2f_k} - 1 = 2e^{f_k} q_k \qquad \ldots(19)$$

$$q_k = \frac{e^{2f_k} - 1}{2e^{f_k}} \qquad \ldots(20)$$

$$q_k = \frac{e^{f_k} - e^{-f_k}}{2} \qquad \ldots(21)$$

Since $\rho_k$ is a parameter, which is defined as

$$\rho_k = f'_k / f'_{k+1} \qquad \ldots(22)$$

$$\rho_k = \frac{\dfrac{1}{\sqrt{1+q_k^2}}}{\dfrac{1}{\sqrt{1+q_{k+1}^2}}} \qquad \ldots(23)$$

$$\rho_k = \frac{1}{\sqrt{1+(\dfrac{e^{f_k} - e^{-f_k}}{2})^2}} \bigg/ \frac{1}{\sqrt{1+(\dfrac{e^{f_{k+1}} - e^{-f_{k+1}}}{2})^2}} \qquad \ldots(24)$$

**4. Sloboda CG-Method:**

The rate of convergence of a variety of CG-algorithm has been investigated by many authors: the most general results have been given by Baptist and Storey (1977) where it was also shown that the algorithms with ELS (exact line searches) have the property of n-step quadratic convergence (Store, 1977). In order to improve the rate of convergence of CG-algorithm it is necessary to construct special algorithms for more general function than the quadratic .In series of papers: Fried (1971); Boland et al., (1979); Tassopoulous and Storey (1984);AL-Bayati et al. (1994),(1995) various algorithms have been suggested which are efficient for special non-quadratic models. Sloboda (1980) first developed an algorithm which generates conjugate direction with inexact line searches and has the same rate of convergence as the classical CG-method without an error vector.

**4.1 Outlines of Sloboda CG-Method:**

**Step1:** $x_0 \in R^n$ , initial point

**Step 2:** set $k=1$; $g_1^- = -g_1$ and $d_1 = g_1^-$

**Step 3:** set $x_{k+1} = x_k + \lambda_k d_k$

**Step 4:** compute $g_{k+1}^+ = g_{k+1}(x - \lambda d / 2)$

**Step 5:** test for convergence: if achieved stop, if not continue

**Step 6:** if $k=n$ or (any equivalent restarting criterion) go to step 3 else continue.

**Step 7:** compute $g_{k+1}^- = \omega g_{k+1}^+ - g_k^-$ where $\omega = d_{k+1}^T g_{k+1}^- / g_{k+1}^T g_{k+1}^+$

**Step 8:** $d_{k+1}$= - $g_{k+1}^- + \beta_k d_k$ where $\beta = g_{k+1}^{-T} g_{k+1}^{-T} / g_k^T g_k$

**Step 9:** set $k=k+1$ and go to Step 3.

**5. Hybrid Conjugate Gradient Methods:**

Despite the numerical superiority of Polak-Riebiere (P/R) algorithm over Fletcher-Reeves (F/R) algorithm, the later has better theoretical properties than the former. Under certain conditions F/R-method can be shown to have global convergence with exact line search (Powell, 1986) and also with inexact line search satisfying the strong Wolfe-Powell condition. (see Al-Baali, 1985).

Normally this leads to speculation on the best way to choose $\beta_k$. Touati-Ahmed and Storey (1995) proposed the following hybrid algorithm:

**Step1:** If $\hat{\zeta}\|g_{k+1}\|^2 \leq (2\zeta)^{k+1}$ , with $\frac{1}{2} > \zeta > \xi$ and $\hat{\zeta} > 0$, go to step 3.

Otherwise, set $\beta_k = 0$.

**Step2:** If $\beta_k^{P/R} < 0$ set $\beta_k = \beta_k^{F/R}$. Otherwise go to step 3.

**Step3:** If $\beta_k^{P/R} \leq (\frac{1}{2}\zeta)\|g_{k+1}\|^2 / \|g_k\|^2$ with $\zeta > \xi$, set $\beta_k = \beta_k^{F/R}$.

Otherwise set $\beta_k = \beta_k^{P/R}$.

Here $\xi$, $\zeta$ and $\hat{\zeta}$ are user supplied parameters. This hybrid was shown to be globally convergent under both exact and inexact line searches and to be quite competitive with P/R-algorithm and F/R-algorithm. See Hu and Storey ( 1991 ) .

Touati and Storey suggested also the following algorithm to compute the conjugancy coefficient $\beta_k$ :

**Step1**: If $\beta_k^{P/R} < 0,$ then $\beta_k = \beta_k^{F/R}$ , return to main program. Otherwise go to step2.

**Step2**: If $0 \le g_{k+1}^T g_k \le \|g_{k+1}\|^2$ , then $\beta_k = \beta_k^{P/R}$ ; return to main program. Otherwise, go to step3.

**Step3**: If ( $\cos^2(\theta_k) \ge \iota_k^2; \quad where \quad \iota_k^2 = \tau / [\|g_k\|^2 \sum_{i=1}^{k} \|g_i\|^{-2}]$ holds, then $\beta_k = \beta_k^{P/R}$ ; return to main program. Otherwise, set $\beta_k = \beta_k^{F/R}$ ; return to main program.

**6. New Suggestion for Hybrid CG-Methods:**

In this section we are going to study develop a new CG-method based on quadratic and non- quadratic models; taking the idea of exact and inexact line searches .The new technique use new hybrid idea between the standard F/R CG-algorithm and Sloboda (1980) CG-method.

**6.1 Outlines of the New Suggested Algorithm:**

**Step1:**Set $x_0 \in R^n$ ,initial point

**Step 2:**Set $k = 1$

**Step3:**Set $d_k = -g_k$

**Step 4:**Set $x_{k+1} = x_k + \lambda_k d_k$

**Step 5:**Check for convergence i.e ,if $\|g_{k+1}\| \prec \varepsilon$ ,then stop else

**Step 6:** Find
$$\rho_k = \frac{1}{\sqrt{1+(\frac{e^{f_k} - e^{-f_k}}{2})^2}} \Big/ \frac{1}{\sqrt{1+(\frac{e^{f_{k+1}} - e^{-f_{k+1}}}{2})^2}}$$

**Step 7:** if $\frac{d_k^T{}_k \, g_{k+1}}{\| g_{k+1} \|} \langle 0.2$ is satisfied go to step(8) otherwise go to step (13)

**Step 8:** $g_k = g(x_k)$

$g_{k+1} = g(x_{k+1})$

**Step 9:**Find $\beta_k = \|g_{k+1}\|^2 / \|g_k\|^2$

**Step 10:** check if $0 \prec \beta_k \prec 1$

**Step 11:** compute $d_{k+1} = -g_{k+1} + \beta_k d_k$, go to step 19
else

**Step 12:**compute $d_{k+1} = -g_{k+1} + \rho_k \beta_k d_k$, go to step 19

**Step 13:** find $g^{+}_{k+1} = g(x - \lambda d / 2)$

**Step 14:** compute $g^{-}_{k+1} = \varpi g^{+}{}_{k+1} - g_k$,where $\varpi = d^T_{k+1}\bar{g}_{k+1} \;/\; g^{+}_{k+1} d^T_{k+1}$

**Step15:** Find $\beta_k = \|g^{-}{}_{k+1}\|^2 / \|g^{-}_k\|^2$

**Step 16:** check if $0 \prec \beta_k \prec 1$

**Step 17:**compute $d_{k+1} = -g^{-}{}_{k+1} + \beta_k d_k$,go to step19
else

**Step 18:**compute $d_{k+1} = -g^{-}{}_{k+1} + \rho_k \beta_k d_k$

**Step 19:**if $k = n$ or (any equivalent restarting critenion)go to step (2),
else

**Step 20:** set, $k = k + 1$and go to step (3)

**7. Numerical Results:**

In order to assess the performance of the new proposed algorithm (Hybrid model). Three minimization algorithms are tested over (10) non-linear unconstrained test functions with different dimensions see (Appendix).

All the results are obtained using (Pentium computer). All programs are written in FORTRAN language and for all cases the stopping criterion taken to be

$$\|g_{k+1}\| \prec 1 \times 10^{-5}$$

Algorithms in this chapter use ELS and ILS strategy which is the cubic fitting technique fully described in (Bunday, 1984).

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

The algorithms are:

1- Standard F/R CG-algorithm.

2- Sloboda CG-algorithm.

3- New Hybrid model algorithm.

In Table (1) we represent comparison between new algorithm with Standard F/R CG-algorithm and Sloboda CG-algorithm.
Our numerical results, which are presented in Table (2) confirm that the Hybrid model algorithm is superior to both Standard CG-algorithm and Sloboda CG-algorithm with respect to the total number of NOF and NOI.

**Table (1)**
**Comparison our new algorithm with standard F/R CG-algorithm and Sloboda CG-algorithm.**

| Test fn. | Dim | F/R CG ALGORITHM NOI(NOF) | Sloboda CG ALGORITHM NOI(NOF) | NEW ALGORITHM NOI(NOF) |
|---|---|---|---|---|
| Powell | 4 | 79(193) | 60(150) | 50(107) |
| Osp | 4 | 8(40) | 6(30) | 6(30) |
| Cubic | 4 | 15(42) | 19(46) | 15(41) |
| Shallow | 4 | 11(26) | 10(26) | 10(26) |
| Cantreal | 4 | 40(269) | 38(209) | 37(246) |
| Osp | 100 | 48(144) | 47(150) | 47(142) |
| Beale | 100 | 57(118) | 69(142) | 62(128) |
| Cubic | 100 | 49(106) | 67(184) | 17(41) |
| Recip | 100 | 8(22) | 8(22) | 11(30) |
| Strait | 100 | 9(20) | 12(26) | 11(24) |
| Sum | 100 | 15(80) | 17(86) | 16(89) |
| Strait | 500 | 9(20) | 12(26) | 15(32) |
| Shallow | 500 | 20(46) | 18(43) | 21(46) |

| | | | | |
|---|---|---|---|---|
| Recip | 500 | 8(22) | 10(29) | 9(24) |
| Sum | 500 | 26(144) | 25(136) | 22(112) |
| Total | | 402(1292) | 418(1305) | 349(1118) |

## ***REFERENCES***


[1] Al-Assady, N.H, and AL-Mashhdany, H. (1997) "**A Rational Logarithmic Science Model for Unconstrained Non-Linear Optimization**", Rafideen Science, J. Vol 8,No.2, pp. 107-17

[2] Al-Assady, N.H and Al-Ta'ai B.A. , (2002) "**A New Non-Quadratic Model For Unconstrained Non-Linear Optimization**" Journal of Al-Raffadin , Computer and Mathematical Science, No 48 , pp.150-151.

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

[4] 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.1 , pp. 133-155 .

[5] AL-Bayati, A.Y. and AL-Assady , N.H. (1994) A.Y, "**Minimization of quadratic functions with inexact line search**, journal of optimization Theory and Applications, Vol. 32, pp.139-147.

[6] AL-Bayati, A.Y. and AL-Naemi , G.M., (1995),"**A combined ECGV M method for nonlinear unconstrained optimization",** Journal of Education and Science, Mosul University ,IRAQ, Vol.23, P.P. 207-214.

[7] 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 .

[8] Baptist, P. and Store, J. (1977), "**On the relation between quadratic termination and convergence properties of minimization algorithms, part 11, Numerical Mathematics**" No.28, pp. 367-391.

[9] Boland W.R.;kamgnia,E.R.and Kowalik, J.S. ( 1979 ) "***Extended CG-method with Restarts"*** journal of optimization theory and application, Vol. 28,pp.1-9 .

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

[11] Dixon, L.C.W. (1975) "***Conjugate Direction Without Linear Search***" Journal of Institute Mathematics and it is Applications 11. PP: 317-328.

[12] Fletcher, R. and Recves , (1964) , "**Function minimization by conjugate gradient**" Compreter . J. 7, pp. 147 – 154.

[13] Fried, I.(1971) "**N-step Conjugate Gradient Minimization Scheme for Non-Quadratic Function**" AIAA jouranl, Vol. 19.pp.2286-2287.

[14] Hestenes, M.R. and Stiefel E. (1952) "**Methods of Conjugate Gradients for Solving Linear Systems"** Journal of Research of the National Burean of Standards, Vol. 49, pp:409-436.

[15] Hu,Y. and Storey, C., (1991) "**Efficient Generalized Conjugate Gradient Algorithms",** Part 1, Theory, Journal of Optimization Theory and Applications, Vol. 69, PP.129-137.

[16] Polak , E. (1969) **"Computational Methods in Optimization"** A Unified Approach , Academic Press , New York.

[17] Powell, M. J. D. (1986) **"How Bad are the BFGS and DFP Method When the Objective Function in Quadratic"** Math programming, Vol. 34, pp.34-47. London.

[18] Sloboda, F ( 1980 ) **"A generalized conjugate gradient algorithm Numerical Mathematics" ,**No. 43 pp. 223-230 .

[19] Store J. (1977) **"On the relation between quadratic termination and convergence properties of minimization algorithms part I. Theory, Numerical Mathematics", Vol.** 28, pp. 343-366**.**

## Appendix

1. Generalized Osp (Oren and Spedicato Function:

$$f(x)=\left[\sum_{i=1}^{n} i x_i^2\right]^2, \qquad x_0=(1,\ldots)^T.$$

2. Generalized Cantreal Function:

$$f(x)=\sum_{i=1}^{n/4}\left[\begin{array}{l}(exp(x_{4i-3})-x_{4i-2})^4+100(x_{4i-2}-x_{4i-1})^6+\\ arctan(x_{4i-1}-x_{4i}))^4+x_{4i-3}\end{array}\right],$$

$x_0=(1,2,2,2;\ldots)^T$

3. Generalized Recip Function:

$$f(x)=\sum_{i=1}^{n/3}\left[(x_{3i-1}-5)^2+x_{9i-1}^2+\frac{x_{3i}^2}{(x_{3i-1}-x_{3i-2})^2}\right], \qquad x_0=(2,5,1;\ldots)^T.$$

4.Generalized Powell Function:

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

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

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

5. Generalized Cubic Function:

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

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

6.Generalized Beale Function:

$$f(x)=\sum_{i=1}^{n/2}\left\{\begin{array}{l}\left[1.5-x_{2i-1}(1-x_{2i})\right]^2+\left[2.25-x_{2i-1}(1-x_{2i}^2)\right]^2\\ +\left[2.625-x_{2i-1}(1-x_{2i}^2\right]^2\end{array}\right\},$$

$x_0=(-1,1,\ldots)^T.$

7.Generalized Shallow Function:

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

$x_0=(-2, -2; \ldots)^T$.

8.Generalized Strait Function:

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

$x_0=(2, -2; \ldots)^T$.

9.Sum of Quatrics (SUM) function:

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

$x_0=(1, \ldots)^T$.