Combined Extended CG Algorithm With Self-Scaling (VM) Updates in Unconstrained Optimization

Adham A. Ali , Fatima Z. A

College of Science , Kirkuk University , Kirkuk , Iraq

(Received: 9 / 10 / 2011 ---- Accepted: 15 / 2 / 2012)

Abstract

In this paper, new combined extended conjugate gradient and variable metric method is proposed for unconstrained optimization. This method is based on exact line searches and compared numerically with standard CG algorithms using a number of test functions with different dimensions, it also obtains encouraging numerical results.

1. Introduction:

The Conjugate Gradient method is an effective method for symmetric positive definite systems. It is the oldest and best known of the nonstationary methods discussed here. The method proceeds by generating vector sequences of iterates (*i.e.*, successive approximations to the solution), residuals corresponding to the iterates, and search directions used in updating the iterates and residuals. Although the length of these sequences can become large, only a small number of vectors needs to be kept in memory. In every iteration of the method, two inner products are performed in order to compute update scalars that are defined to make the sequences satisfy certain orthogonality conditions. On a symmetric positive definite linear system these conditions imply that the distance to the true solution is minimized in some norm, [16].

To summarize the standard CG-algorithm as follows:

<u>Standard CG Algorithm:</u>

Step (1)= Set $x_0 \in R^n$, ε, *n* (initial point, scalar, termination).

Step (2): for *k=1*, $d_k = -g_k$

Step (3): Compute $x_k = x_{k-1} + \lambda_{k-1} d_{k-1}$

where λ_{k-1} is obtained from the line search procedure.

Step (4): Calculate the new direction

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

where β_k is the conjugacy coefficient and it has the following formula:

$$\beta_k^{HS} = \frac{g_k^T (g_k - g_{k-1})}{[d_{k-1}^T (\rho_k g_k - g_{k-1})]} \qquad \text{(calculated by [15])}$$

This form is considered a general form for the classical CG-method.

$$\beta_k^{FR} = (\|g_k\|^2 / \|g_{k-1}\|^2) \qquad \text{(calculated by [13])}$$

$$\beta_k^{PR} = \frac{g_k^T (g_k - g_{k-1})}{(g_{k-1}^T g_{k-1})} \qquad \text{(calculated by [21])}$$

$$\beta_k^{DX} = -\|g_k\|^2 / d_{k-1}^T g_{k-1} \qquad \text{(calculated by [11])}$$

Step (5): Check for convergence

If $\|g_k\| \le \varepsilon$, then stop

Step (6): Check for restarting criterion

If *k = n*, then stop, Otherwise set *k = k +1* and go to step (3).

2.Review of the Extended Conjugate Gradient and Self- Scaling VM Method:

a) Extended conjugate gradient methods (ECG)

The CG-methods discussed so far assume a local quadratic representation of the objective function. However, quadratic models may not always be adequate to incorporate all the information which might be needed to represent the objective function successfully. Also in problems where the quadratic representation is not good. A non-quadratic model may be better represent the objective function and that leads to speculation on a better way to choose a type of a non-quadratic model.

In boland [9], it was observed that q(x) and F(q(x)) have determined the same search direction so that the finite termination property is satisfied.

Many Authors have suggested special models to determine ρ_k where ρ_k is defined as follows:

$$\rho_k = \frac{f'_{k-1}}{f'_k} \qquad \dots (1)$$

Where f' is defined as follows:

$$f = F(q(x)), \frac{dF}{dq} = \bar{F} > 0 \text{ and } q(x) > 0 \quad \dots (2)$$

And F is an increasing monotonic function, which is better represent the objective function and q is a quadratic function.

i) $F(q(x)) = ((q(x))^\rho, \rho > 0 \ x \in R^n$ by [14] … (3)

ii) $F(q(x)) = \varepsilon_1 q(x) + \frac{1}{2}\varepsilon_2 q^2(x)$ by [8] and [9] … (4)

iii) $F(q(x)) = \varepsilon(\log q(x) - 1), \varepsilon > 0$ by [5] … (5)

iv) $F(q(x)) = \exp(\frac{\varepsilon_1 q(x)}{\varepsilon_2 q(x) - 1}); \ \varepsilon_2 < 0$ by [6]… (6)

v) $F(q(x)) = \coth^{-1}(\varepsilon q(x))$ by [1] … (7)

Now we are going to list down the outline of the extended CG-algorithm by using model [1] .

<u>(ECG) Algorithm:</u>

Given $x_0 \in R^n$ the initial point, and scalar ε .

Step (0): Set $d_\circ = -g_\circ$

Step (1): For *k = 1,2,...*

Compute $x_k = x_{k-1} + \lambda_{k-1} d_{k-1}$

where λ_{k-1} is the minimizer of f on d_{k-1}.

Step (2): Check for convergence

If $\|g_k\| \le \varepsilon$, then stop, Otherwise continue.

Step (3): Calculate

$$Exp(f) = f + f' + \frac{f^{(2)}}{2!} + \frac{f^{(3)}}{3!} + \ldots$$

Step (4): Compute

$$\rho_k^{New} = \left[\frac{e^{f_{k-1}} \_ e^{-f_{k-1}}}{e^{f_k} - e^{-f_k}}\right]^2$$

Step (5): Check if $0 \le \rho_k \le 1$, then go to step(6). Otherwise set $\rho_k = 1$ and go

to step(6).

Step (6): Calculate the new direction

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

where β_k is defined in section (1)

Step (7): Check for restarting criterion

If *k = n*, then stop, Otherwise set *k = k +1* and go to step (1).

b) Self-Scaling VM Method

To alleviate the family of VM updating, it is useful to multiply each H_k by some scale factor $\rho_k > 0$ before using the update formula with exact line searches, this can be shown to present the conjugacy property in the quadratic case, although we may no longer have $H_K \equiv G^{-1}$. However, the focus here is to improve the single-step rather than n-step convergence behavior of the algorithm. Methods that automatically prescript scale factor in a manner such that, if the function is quadratic then the eigenvalues of $d_k^T H_k G_{K+1}$ tend to be spread above and below are called self-scaling methods [7].

In the 1970's the self-scaling VM algorithms were introduced, showing significant improvement in efficiency over standard VM-methods. In particular, in a series of papers by Oren, [17], Oren and Luenberger, [18], Oren and Spedicato, [19], Al-Bayati, [2], Al-Bayati and Al-Salih, [4], algorithms for minimizing an unconstrained non-linear function f(x) were developed. Given an initial estimate x_1 to the minimizer x^* and with numerical estimate H_1 if the inverse Hessian matrix $G^{-1}(x)$. VM algorithms generate a sequence of points x_k by:

$$x_{k+1} = x_k - \lambda_k H_{k+1} g_{k+1} \qquad \ldots (8)$$

And H_1 is updated by:

$$H_{k+1} = H_k - \frac{H_k y_k y_{ki}^T H_k}{y_k^T H_k y_k} + \phi_i w_k w_k^T + \theta_k \frac{v_{ki} v_{ki}^T}{v_k^T y_k} \quad \ldots (9a)$$

With

$$w_k = (y_k^T H_k y_k)^{\frac{1}{2}} \left[\frac{v_k}{(v_k^T) y_k} - \frac{H_k y_k}{y_k^T H_k y_k}\right] \ldots (9b)$$

Where ϕ_i , θ_i are scalers. The updating is performed so that to satisfy the QN condition

$$H_{k+1} y_k = \theta_k v_k \quad \ldots (10)$$

This condition is commonly satisfied with $\theta_k = \theta = 1$, with this restriction on (9) we have the (Broyden family) of algorithms see [12]. The most successful version of Broyden family is BFGS (Broyden, Fletcher, Goldfarb, and Shanno), which corresponds to a choice of $\phi_K = 1$.

We can now summarize the scaled BFGS algorithm due to Al-Bayati,[2]:

<u>(AL-Bayati) Algorithm:</u>

Start with an initial point x_1

Step (1): set *k=1* and choose $H_1 = I$, where *I* is the identity matrix.

Step (2): Determine the step-size λ_k that minimizes $f(x_k + \lambda d_k)$ where

$d_k = -H_k g_k$, and obtain $x_k = x_{k-1} + \lambda_{k-1} d_{k-1}$

Step (3): Update H_k by H_{k+1} by using Al-Bayati's update as follows:

$$H_{k+1}^{lA\wp-Bayati} = H_k - \left\{I - \frac{v_k y_k^T}{v_k^T y_k}\right\} H_k \left\{I - \frac{y_k v_k^T}{v_k^T y_k}\right\}$$

$$+ \sigma_k \frac{v_k v_k^T}{v_k^T y_k} + \phi_k w_k w_k^T$$

…(11a)

Where $\phi_k = 1$, and $\sigma_k = \frac{y_k^T H_k y_k}{v_k^T y_k}$ … (11b)

And w_k is a vector defined earlier in (9b).

Step (4): set *k=k+1* and repeat.

Self-Scaling VM algorithms possesses the following properties for a quadratic function:

i) Since all d_k are mutually conjugate (with respect to G) in self-scaling VM algorithms, then the solution is obtained in at most n iterations.

ii) If $\lambda_k = 1$ for all k, then the algorithm converges (two-step super linearly), i.e.

$$\lim_{k \to \infty} \frac{\|x_{k+1} - x^*\|}{\|x_k - x^*\|} = 0 \qquad \ldots (12)$$

iii) They are globally convergent.

The proofs of these properties can be found in (Al-Bayati, [2], and Oren, [20]).

3. <u>New Combined Extended CG and Self-Scaling VM Method:</u>

Both CG and VM methods are relatively successful at minimizing smooth non linear functions of several variables. In particular CG-methods require less storage to implement than VM algorithms and are therefore preferred when storage limitations occur. However, they have a slower rate of convergence, so there have been some attempts to combine CG and VM algorithms in order to obtain good convergence properties and low storage requirements see [12].

The most successful algorithms in this field are: shanno and Phua, [22]; and Al-Bayati, [3].

In this section, we shall describe a new algorithm which effectively interleave the extended CG-algorithm in [1] with the Al-Bayati's [2] Self-Scaling VM steps.

New Combined Algorithm:

```
Step (1): set k=1 and $d_k = -H_k g_k$.
Step (2): do a line search: $x_k = x_{k-1} + \lambda_{k-1} d_{k-1}$
Step (3): If $\|g_k\| \le \varepsilon$, then stop, Otherwise continue.
Step (4): if $k>n$ or $d_{k+1}^T g_{k+1} > -0.8\|g_{k+1}\|$ is satisfied then go to step(8)
Step (5): compute
    $\rho_k^{New} = \left[\frac{e^{f_{k-1}} - e^{-f_{k-1}}}{e^{f_k} - e^{-f_k}}\right]^2$
Step (6): set the new direction
    $d_{k+1} = -H_k g_{k+1} + \beta_k d_k$
    where $\beta_k$ is defined by:
    $\beta_k^{HS} = \frac{g_k^T H_k(\rho_k g_k - g_{k-1})}{[d_{k-1}^T(\rho_k g_k - g_{k-1})]}$    (Hestenes-Stiefel [15])
Step (7): set k=k+1 and go to step(2)
Step (8): update $H_k$ by using Al-Bayati's update in equation (11)
Step (9): set $x_k = x_1$ and go to step(1)
```

4. Numerical Computation

The comparison tests involve thirty well-known test functions with different dimensions (see Appendix). All the results were obtained using (Pentium 4 computer) using programs written in "FORTRAN". The comparative performance of the algorithms are evaluated by considering both the total number of iterations (NOI) and total number of function evaluations (NOF). The stopping criterion is taken to be:

$$\|g_{k+1}\| \le 5\times10^{-5}$$

The line search routine employed is the cubic fitting technique which uses function values and their gradients. The actual program used is an adoption of the routine published in [10].

Two algorithms were tested, namely

1) Standard CG-algorithm using H/S formula

2) New combined algorithm

To solve thirty test functions with dimensionally varies between $2 \le n \le 400$.

Our numerical results are presented in the following two tables.

Table (1) compares between the standard CG-algorithm and the new combined algorithm for small dimensionally test functions $2 \le n \le 80$. Fifteen non-linear test problems are used for this purpose. While Table (2) deals with the large dimensionality test functions $100 \le n \le 400$, with the same sort of comparison.

It is clear that the new algorithm out perform the standard CG-algorithm as a result of this comparison.

Table (1)

Test Function	n	H/S	
		New NOI(NOF)	Standard CG NOI(NOF)
Powell	40	46(128)	72(158)
Cantreal	4	29(192)	33(230)
Rosen	4	17(44)	27(72)
Cubic	4	13(32)	16(42)
shallow	2	28(67)	48(101)
Wolfe	80	34(69)	49(99)
Bigg	40	29(65)	43(141)
OSP	10	9(42)	10(48)
Cantreal	20	18(124)	18(123)
Wood	20	43(72)	52(107)
Dixon	2	27(39)	32(67)
OSP	50	20(69)	24(79)
Rosen	80	24(56)	23(56)
Wood	40	50(102)	48(101)
Dixon	10	18(36)	22(46)
Total NOI(NOF)		405(1137)	517(1470)

Table (2)

Test Function	n	H/S	
		New NOI(NOF)	Standard CG NOI(NOF)
Powell	100	89(193)	129(263)
Cantreal	200	15(79)	19(137)
Rosen	100	16(37)	23(56)
Cubic	100	10(25)	14(37)
OSP	100	15(54)	17(67)
Bigg	100	29(65)	45(142)
Wood	100	52(107)	85(175)
Dixon	100	75(702)	120(860)
Shallow	200	5(13)	6(17)
Bigg	200	29(65)	29(65)
Rosen	300	12(29)	24(59)
Wolfe	300	31(62)	53(107)
Powell	300	183(389)	398(661)
Cubic	400	9(25)	12(35)
Wood	400	67(121)	86(176)
Total NOI(NOF)		1756(4137)	3034(6737)

Indeed, it is clear from table (1) that taking the standard CG-algorithm as 100% NOI; NOF yields:

Table (3)

Measure	Standard H/S-CG	New
NOI	100%	78%
NOF	100%	77%

It is obvious that new combined algorithm improves the standard CG-algorithm in about 22% NOI: 23% NOF.

Now from table (2) taking the standard CG-algorithm as 100% NOI; NOF yields:

Table (4)

Measure	Standard H/S-CG	New
NOI	100%	71%
NOF	100%	67%

It is obvious that new combined algorithm improves the standard CG-algorithm in about 29% NOI: 33%

NOF. This will agree the fact that the Self-Scaling VM-updates will improve the behavior of CG-method for high dimension test function.

5. Appendix (Test functions)

1- Bigg Function:

$$f=\sum_{i=1}^{10}\begin{pmatrix}\exp(-x_1z_i)-5\exp(-x_2z_i)-\exp(z_i)+\\ 5\exp(-10z_i)\end{pmatrix}^2,\ z_i=\frac{i}{10}$$

$$x_\circ=(1,2)^T$$

2-Cubic Function:

$$f=100(x_2-x_1^3)^2$$

$$x_\circ=(-1.2,1)^T$$

3- Dixon Function:

$$f=(1-x_1)^2+(1-x_{10})^2+\sum_{i=1}^{9}(x_i^2-x_{i+1})^2$$

$$x_\circ(-1,-1)^T$$

4-Generalized Cantreal Function:

$$f=\sum_{i=1}^{\frac{n}{4}}[\exp(x_{4i-3})-x_{4i-2}]^2+100(x_{4i-2}-x_{4i-1})^6$$

$$+[[a\tan(x_{4i-1}-x_{4i})]^4+x_{4i-3}^8]$$

$$x_\circ=(1,2,2,2)^T$$

5-Oren and Spedicato Power Function (OSP):

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

$$x_\circ=(1,\ldots)^T$$

6-Rosenbrock Function:

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

$$x_\circ=(-1.2,1)^T$$

7-Generalized Powell Function:

$$f=\sum_{i=1}^{n/4}\begin{bmatrix}(x_{4i-3}-10x_{4i-2})^2+5(x_{4i-1}-x_{4i})^2\\ +(x_{4i-2}-2x_{4i-1})^4+10(x_{4i-3}-x_{4i})^4\end{bmatrix}$$

$$x_\circ=(3,-1,0,1,\ldots)^T$$

8-Wolfe Function:

$$f=\left[-x_1(3-\frac{x_1}{2})+2x_2-1\right]^2+\sum_{i=1}^{n-1}\left[x_{i-1}-x_i(3-\frac{x_i}{2})+2x_{i-1}-1\right]^2+\left[x_{n-1}-x_n(3-\frac{x_n}{2})-1\right]^2$$

$$x_\circ=(-1,\ldots)^T$$

9-Shallow Function:

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

$$x_\circ=(-2,-2,\ldots)^T$$

10-Wood Function (generalized form):

$$f=\sum_{i=1}^{\frac{n}{4}}\begin{bmatrix}100(x_{4i-2}-x_{4i-3}^2)^2+(1-x_{4i-3})^2+90(x_{4i}-x_{4i-1}^2)^2+(1-x_{4i-1})^2\\ +10.1((x_{4i-2}-1)^2+(x_{4i}-1)^2)+19.8(x_{4i-2}-1)(x_{4i}-1)\end{bmatrix}$$

$$x_\circ=(-3,-1,-3,-1,\ldots)^T$$

6. References:

[1] Adham, A. Ali. (2011), "A New Non Quadratic Algorithm For Solving Non-Linear Optimization Problems", journal of kirkuk university-scientific studies, Vo 6. No 2, PP. 201-212.

[2] Al-Bayati, A. Y. (1991), "A New Family of Self-Scaling Variable Metric Algorithms for Unconstrained Optimization", Journal of Education on Science, Mosul University, 12,pp.25-54.

[3] Al-Bayati, A. Y.(1994), "A New Interleaved VM-CG Algorithm for Unconstrained Optimization", Journal of Education on Science, Mosul University, vol(21),pp.159-168.

[4] Al-Bayati, A. Y. and M. S. Al-Salih (1994), "New Variable Metric Methods for Non-Linear Unconstrained Optimization", Journal of Education and Science Mosul, IRAQ, Vol (21), pp.169-182 .

[5] Al-Bayati, A. Y. (1995), "New Extended CG. Methods for Non Linear Optimization", Natural and Applied Series, Mu'tah Journal for Research and Studies. Mu'tah University, Jordan Vol. 10. No. 6, PP. 69-87.

[6] Al-Mashhadany, H.(2002), "A Rational Exponential Model for Unconstrained Non-Linear Optimization", Qatar University .Sc.J,pp.5-16

[7] Bazaraa, H. D. Sherali and C. M. Shetty(1993), "Non-Linear Programming Theory and Algorithms". NewYork, Chichester, Brisbane, Toronto.

[8] Boland, W. R. Kamagnia, E. R. and Kowalik, J. S. (1977a), "A Conjugate Gradient Optimization Method in Variant to Non-Linear scalling", Journal of Optimization Theory and Application, 27, PP. 221-230.

[9] Boland, W. R. and kowalik, J. S. (1979b), "Extended Conjugate Gradient Method with Restarts", Journal of Opt. Theory and Application, 28, PP. 1-9.

[10] Bunday, B. (1984), "Basic Optimization Methods", Edward Arnold, Bedford Square, London, (1984).

[11] Dixon, L. C. W. (1972), "*Non-Linear Optimization*", English University Press, London.

[12] Fletcher R.(1987), "Practical Methods of Optimization" John Wiley and Sons ,Chichester . New York.

[13] Fletcher, R. and Revees. (1964), "Function Minimization by Conjugate Gradient", Computer J. 7, 147-154.

[14] Fried, I. (1971), 'N-Step Conjugate Gradient Minimization Scheme for Non - Quadratic Function", AIAA J., a.

[15] Hestences, M. R. and Stiefle, E. (1952), "Methods of Conjugate Gradient for Solving Linear Systems", J. Res. N. B. S., 49.

[16] Jack Dongarra. (1995)) "Conjugate gradient methods (CG)" http://www-db.stanford.edu/TR/CS-TR-90-1330.html

[17] Oren, S. S. (1974), "Self-Scaling Variable Metric Algorithms, Part II", Management Science,20, PP. 863-874.

[18] Oren, S. S. and Leunberger, D. (1974), "Self-Scaling Variable Metric Algorithm", Part I. Management Science, 20, PP. 845-862.

[19] Oren, S. S. and Spedicato, E. (1976), “Optional Conditioning of Self-Scaling VM Algorithms”, Mathematical Programming, 10, PP. 70-90.

[20] Oren, S. S. (1979), “Self-Scaling Variable Metric Algorithm without Line Search for Unconstrained Minimization”, Mathematics of Computation 27.

[21] Polak, E. and Ribier, G. (1969). “Note Surla Convergence Des Methods De Directions Conjugate”, Rev. Fr. infr, Rech Oper, 16, R.

[22] Shanno, D. F and Fhua, K.H. (1978), “Matrix Conditioning and Non-Linear Optimization”, Math. Prog. 14, pp.149-160.

ربط خوارزمية التدرج المترافق الموسع مع تحديث المتري المتغير ذاتي القياس في الامثلية غير المقيدة

ادهم عبدالوهاب علي ، فاطمة زين العابدين احمد

كلية العلوم ، جامعة كركوك ، كركوك ، العراق

(تاريخ الاستلام: 9 / 10 / 2011 ---- تاريخ القبول: 15 / 2 / 2012)

الملخص

في هذا البحث تم استحداث خوارزمية جديدة تداخلية تربط خوارزمية التدرج المترافق الموسع مع خوارزميات المتري المتغير ذاتي القياس في الامثلية غير المقيدة، الخوارزمية الجديدة تستعمل خط البحث التام. الحسابات العددية أثبتت كفاءة الخوارزمية الجديدة مقارنة مع خوارزمية CG القياسية.