# Modified Preconditioned Conjugate Gradient Algorithms for Lagrange method

***Eman Tariq Al-Haj saeed *** ***Huda Issam Ahmed****
***Aseel Muayad Qasim*****

Received 20, December, 2012
Accepted 11, March, 2014

## Abstract:

This paper is cornered with the improvement of conjugate gradient type methods (CG methods) to solve non-linear constrained optimization problems by using Lagrange method. Most numerical algorithms are sensitive to error due to arithmetic operation , therefore we suggested the self scaling conjugate Gradient method to avoid this difficulty and to increase the ability for solving algorithms for ill- problems. Our new modified CG method shows that , it is too effect when compared with other established algorithms to solve standard constant optimization problems.

**Keywords: conjugate gradient, non-linear constrained optimization, Lagrange method.**

## 1. Introduction to constrained method

The general optimization problem can be stated as follows:

$$\underset{x\in R^n}{\text{Minimize}}\ f(x)\ \text{ subject to}\begin{Bmatrix} h_k(x)=0, & k\in\varepsilon \\ c_k(x)\geq 0, & k\in\tau \end{Bmatrix} \quad (1)$$

where f and the functions $c_k, h_k$ are all smooth, real-valued functions on a subset of $R^n$, and $\varepsilon$ and $\tau$ are two finite sets of indices. We call f the objective function , $h_k, k\in\varepsilon$ are the equality constraints and $c_k, k\in\tau$ are the inequality constraints .We define the feasible set $\psi$ to be the set of points x that satisfy the constraints; i.e., $\psi=\{x: h_k(x)=0,\ k\in\varepsilon\}$ when the constrained is equal zero, $\phi=\{x:\ c_k(x)\geq 0, k\in\tau\}$ when the constrained is grater than zero , so that we can rewrite (1) more compactly as a new objective function $L(x_k,\lambda)$ such that:

$$L(x_k,\lambda)=f(x_k)+\lambda_k\sum_{i=1}^{m}\Psi\left[h_i(x_k)\right]+\lambda_k\sum_{i=m+1}^{n}\phi\left[c_i(x_k)\right] \quad (2)$$

where $\lambda_k$ are Lagrange multipliers and the remainder of second term is the Lagrange function , ( see [1] ).

## 2. The Preconditioned Conjugate Gradient methods

In the application of the general method the Quasi- Newton (QN) and Conjugate gradient (CG) methods each has particular advantages and disadvantages. McCormick and Ritter ( see [2] ) showed that, in general, the QN methods converge faster (and require fewer function evaluations) than the CG methods. However, QN methods normally construct a sequence of symmetric positive definite matrices of order n, so it is necessary to have n(n+1)/2 locations in a high speed computer

*College of Computer Sciences and Mathematics University of Mosul , Iraq
**College of Education, University of Mosul , Iraq

storage. As n increases, n(n+1)/2 must become too large and the variable metric methods cannot be used efficiently only with high memory.

For this reason a new class of CG methods has been developed, termed preconditioned conjugate gradient method (PCG) . Its aim is to keep the storage requirements of order n while improving the convergence properties, (see [3],[4]) . This method is used for solving systems of linear equations, (see [5]). The idea of preconditioning has been extended directly to nonlinear problems.

The relative advantages of methods of this type is that they require less storage and computation time and they are not so sensitive to the exactness of the line search. Recently, several papers have been suggested to the CG method by introducing a preconditioned matrix H (see [6]).

## 3. A Scaled Preconditioned Conjugate Gradient method

This paper is concerned with self-scaling conjugate gradient algorithms for finding a local minimum of the constrained optimization problem, These algorithms start with an initial approximation $x_1$, of a solution, $x^*$; and generate new approximations by the basic iteration:

$$x_{k+1} = x_k - \eta_k H_k \nabla f(x_k) \quad \ldots(3)$$

where $\eta_k$ is a step length (calculated by line search), and $H_k$ is an approximation to the inverse Hessian matrix $G^{-1}$. Given an initial symmetric and positive definite $H_1$, new Hessian approximations are generated by the two-parameters family of updates:

$$H_{k+1} = \gamma\left(H_k - \frac{H_k y_k y_k^T H_k}{y_k^T H_k y_k} + \theta\, v_k v_k^T\right) + \left[\frac{s_k s_k^T}{s_k^T y_k}\right] \quad \ldots(4)$$

with

$$\left\{\begin{array}{l} v_k = \sqrt{y_k^T H_k y_k}\left(\frac{s_k}{s_k^T y_k} - \frac{H_k y_k}{y_k^T H_k y_k}\right), \\ s_k = v_{k+1} - v_k \\ y_k = g_{k+1} - g_k \end{array}\right\} \quad \ldots(5)$$

where $\gamma$ and $\theta$ are scaling and updating parameters, respectively. (For background on these updates, (see [7]). If $\gamma = 1$; eq. (4) reduces to the classical (unscaled) Broyden family of Hessian approximation updates. In this case, although for $\theta \in [0,1]$; which is called the convex class, iteration (3) converges globally and q-super linearly for convex objective functions under appropriate conditions (see [8]) , only updates with $\theta = 0$ (which corresponds to the BFGS update) have been shown to be effective in practice; and the performance of the algorithm varies as $\theta_i$ increases from 0 to 1. See in particular, [9], [10], [11], and [12].

Several attempts have been made to improve the performance of the above class of algorithms by choosing $\gamma$ and $\theta$ in such a way to improve the conditioning of $H_{k+1}$ ( see [7]) . In fact, Nocedal and Yuan [13] showed that the best self scaling BFGS algorithm of Oren and Luenberger [7] performs badly compared to the BFGS method when applied with inexact line searches to a simple quadratic function of two variables.

Al-Baali [14], however, used the theory of Byrd, Liu and Nocedal [8] for unscaled methods to determine conditions on $\gamma$ and $\theta$ that ensure global and superlinear convergence for scaling algorithms with inexact line searches under the additional restriction that:

$$\gamma \le 1 \quad \ldots(6)$$

Using these conditions, Al-Baali [1] and [2] showed that the performance of some members of the Broyden family, including the BFGS update, was improved substantially.

Condition (6) is motivated by the fact that the eigenvalues of the bracketed matrix in (4) can be reduced if $\gamma < 1$(even for $\theta = 0$) and, hence, smaller eigenvalues are introduced into $\beta_{k+1}$ if the eigenvalues of $H_k$ are large. On the other hand, since the BFGS update corrects small eigenvalues of $H_i$ (see e.g. [8], [6] and [11]), it is sensible to use $\gamma_k = 1$ if $\theta_k \le 0$.

Oren and Spedicato [15] considered minimizing the condition number of $H_k^{-1}H_{k+1}$ with respect to $\theta$, and derives the relationship:

$$\theta = \frac{b(c - b\gamma)}{\gamma(ac - b^2)}, \quad \ldots(7)$$

where

$a = y_k^T H_k y_k$,

$b = s_k^T y_k$,

$c = s_k^T H_k s_k$.

Now substituting $\theta = 1$ in (7) yields

$$\gamma = \frac{s_k^T y_k}{y_k^T H_k y_k} \quad \ldots(8)$$

Shanno (1978) showed that one can generate a class of CG-algorithms from Broydens class , the step size still has no relation to the actual step size of the minimum, an addition truncation error can be added. In order to develop conjugate gradient method to solve (ill –problems)which is less sensitive to the rounding off error and truncation off error, we have suggested in this paper a self–scaling conjugate gradient method based on shanno ( see [16]) .

## 4. The derivation of A Scaled Preconditioned Conjugate Gradient method in equality constrained optimization

In this section we present a modified conjugate gradient method by employing scaling parameter in order to play an important rule in the convergence behavior of conjugate gradient method. The scalar $\gamma$ is chosen so that to reduce sensitivity of the step sizes of the line search.

Now we derive self- scaling conjugate gradient algorithm as follows:

Let $k = 2,3,...,n$

$$d_k = -g_k + \sum_{i=1}^{k-1} \alpha_i d_i \quad \ldots(9)$$

Set

$$d_2 = -g_2 + \alpha_1 d_1 \quad \ldots(10)$$

Choose $\alpha_1$ which satisfiyes the conjugacy condition (i.e. $d_i^T G d_j = 0,\ i \ne j$) where G is the Hessian matrix, and the matrix H in the constrained problem is define by:

$$\overline{H}_{k+1} = H_{k(BFGS)} + \sum_{i=1}^{m} \lambda_k \phi''[h_i(x)] \nabla h_i(x) \nabla h_i^T(x) \quad \ldots(11)$$

Where $H_{k(BFGS)}$ is defined in equations (4-5), in such a way $d_1$ and $d_2$ are conjugate so we have: $d_1^T \overline{H}_{k+1} d_2 = 0$ (12)

Substitute $d_2$ which defined above in (10) and put it in (12) then we get:

$$-d_1^T \overline{H}_{k+1} g_2 + \alpha_1 d_1^T \overline{H}_{k+1} d_1 = 0 \ldots(13)$$

So we have:

$$\alpha_1 = \frac{g_2^T \overline{H} d_1}{d_1^T \overline{H} d_1} \quad \ldots \quad (14)$$

Since $\overline{g}_k$ is the gradient of constrained problem, then:

$$\alpha_1 = \frac{g_2^T(\overline{g}_2 - \overline{g}_1)}{d_1^T(\overline{g}_2 - \overline{g}_1)} \ldots(15)$$

Let k=3

$$d_3 = -g_3 + \alpha_2 d_2 \quad \ldots(16)$$

For the conjugacy $d_2$ and $d_3$ we get

$$d_2^T \bar{H} d_3 = 0 \quad \ldots(17)$$

Substitute $d_3$ which defined above in eq.(16) and put it in (17) then:

$$-d_2^T \bar{H} g_3 + \alpha_2 d_2^T \bar{H} d_2 = 0$$

…(18)

**From conjugacy condition**

$$\alpha_2 = \frac{g_3^T(\bar{g}_3 - \bar{g}_2)}{d_2^T(\bar{g}_3 - \bar{g}_2)} \quad \ldots(19)$$

In general for all k we have the relation

$$d_{k+1} = -g_{k+1} + \alpha_k d_k + \sum_{j=1}^{k-j-1} l_{k,j} d_{k-1,j-1}$$

…(20)

We must prove

$$l_{k,j} = 0 \qquad m < k-1 \quad \ldots(21)$$

From $d_{k-j-1} < d_{k-1}$ we suppose $k - j - 1 = m$

Then $$l_{k,j} = \frac{g_k^T y_k}{d_k^T y_k} \quad \ldots(22)$$

$$g_k^T y_m = 0 \qquad \text{i,e} \quad g_k^T (g_m - g_{m-1}) = 0$$

For $m < k-1$ then we have $g_k^T g_m$ and $g_k^T g_{m-1} = 0$

The gradient $g_0$ at $x_0$ and the gradient at the minimum $x$ is $g_1$, Will be orthogonal to $g_0$.Then

$$l_{k,j} = 0 \qquad m < k-1 \quad \ldots(23)$$

**Therefore**

$$d_{k+1} = -g_{k+1} + \alpha_k d_k \quad \ldots(24)$$

## 5. The derivation of a new Scaled PCG method:

In this section we modify the direction in eq.(24) by multiplying it with several values of scaling parameter which is based on Oren and Spedicato (1976) idea as follows:

1- $$\gamma_1 = \frac{s_k^T y_k}{y_k^T \bar{H}_k y_k} \quad \ldots(25)$$

2- $$\gamma_2 = -\frac{s_k^T \bar{g}_k}{(\bar{g}_{k+1}^T \bar{H}_k \bar{g}_{k+1}) + (\bar{g}_k^T \bar{H}_k \bar{g}_k)} \ldots(26)$$

3-

$$\gamma_3 = -\frac{s_k^T \bar{g}_k}{(\bar{g}_{k+1}^T H_{k(BFGS)} \bar{g}_{k+1}) + (\bar{g}_k^T H_{k(BFGS)} \bar{g}_k)}$$

…(27)

to derive those scalar we depend on the following properties (orthogonal, exact line search ) and noting that $y_k = \bar{g}_{k+1} - \bar{g}_k$ :

A- from eq.(8) and where $H_{k+1}$ is defined by eq.(11) we get:

$$\gamma_1 = \frac{s_k^T y_k}{y_k^T \left( H_{k(BFGS)} + \sum_{i=1}^{m} \lambda_k \phi''[h_i(x)] \nabla h_i(x) \nabla h_i(x)^T \right) y_k}$$

, so the direction is define by:

$$d_{k+1} = \gamma\left(-\bar{g}_{k+1} + \alpha_k d_k\right)$$

where $$\alpha_k = \frac{g_{k+1}^T(\bar{g}_{k+1} - \bar{g}_k)}{d_k^T(\bar{g}_{k+1} - \bar{g}_k)}$$

…(29)

The above direction was obtained when we use all of the above properties and define $I = \{i : h_i(\bar{x}) = 0\}$, $N = \{i : h_i(\bar{x}) < 0\}$ and the cone $C = \{d \neq 0, \nabla h_i(\bar{x}) d = 0$ for $i \in I$ and $\nabla h_i(\bar{x}) d > 0$ for all $i \in N\}$.

B- Using the properties of CG-method (the orthogonally property $g_{k+1}^T g_k = 0$, and exact search property $g_{k+1}^T d_k = 0$) we get:

$$\gamma_2 = -\frac{s_k^T \bar{g}_k}{(\bar{g}_{k+1}^T \bar{H}_k \bar{g}_{k+1}) + (\bar{g}_k^T \bar{H}_k \bar{g}_k)},$$

C- Using the properties of (B) and taking $I = \{i : h_i(\bar{x}) = 0\}$ then:

$$\left(\sum_{i=1}^{m}\lambda_k\phi''[h_i(x)]\nabla h_i(x)\nabla h_i(x)^T\right)=0,$$

so we get:

$$\gamma_3=-\frac{s_k^T\bar{g}_k}{(\bar{g}_{k+1}^T H_{k(BFGS)}\bar{g}_{k+1})+(\bar{g}_k^T H_{k(BFGS)}\bar{g}_k)}$$

## 6. Outline of the new Scaled PCG algorithm:

step (1): Choose an initial point $x_0$ in the interior of the feasible region for the equality constraint and use an initial positive definite matrix $H_1 = I$.

step (2): set $k=1$.

step (3): solve $d_1 = -H_1 g_1$.

step (4): compute $\eta$ where $\eta$ is a step length ( calculated by line search) and then compute $x_{k+1}=x_k+\eta_k d_k$.

step (5): check for convergence if $\|g_{k+1}\|<1\times10^{-5}$ is satisfied then stop, else, continue.

step (6): compute the new search direction :

$d_{k+1}=\gamma(-g_{k+1}+\alpha_k d_k)$, where $\gamma$ is define by (25 or 26 or 27) and $\alpha_k$ is defined by ( $\alpha_k=\frac{g_{k+1}^T(\bar{g}_{k+1}-\bar{g}_k)}{d_k^T(\bar{g}_{k+1}-\bar{g}_k)}$ ).

Step (7): check for restarting criterion, i.e. if $d_{k+1}^T g_{k+1}\geq -0.8\|g_{k+1}\|^2$ is satisfied go to step 2 else, set k=k+1, $\lambda_{k+1}=\lambda_k+\sum_{i=1}^{m}\Psi[h_i(x_k)]$ go to step 4.

## 7- Numerical Results and Conclusion:

The basic idea of the new proposed algorithm is to combine the scaled memory less BFGS method and the preconditioning technique. The preconditioned, which is also a scaled BFGS matrix, is reset when a restart criterion holds. Therefore , we get a preconditioned and scaled BFGS algorithm of constrained optimization problem.

We compare the performance of the several new proposed scaled PCG algorithm with the original CG algorithm on a set of constrained optimization problem.

All programs are written in FORTRAN language and for all cases the stopping criterion taken to be $\|g_{k+1}\|<1\times10^{-5}$

All the algorithms in this paper use the same ELS strategy which is the quadratic interpolation technique directly adapted from Bunday [ 17] .

The comparative performance for all of these algorithms are evaluated by considering numbers NOF, NOI, and NOC where NOF is the number of function evaluations, NOI is the number of iterations and NOC is the number of constrained evaluations, is considered as the comparative performance of the following algorithms:

**1- standard CG-algorithms**

**2- self -scaling CG-algorithms(New 1) defined in equation (25)**

**3- self -scaling CG-algorithms(New 2) defined in equation (26)**

**4- self -scaling CG-algorithms(New 3) defined in equation (27)**

Our numerical results were presented in the following table (1) which confirms that the self-scaling CG-algorithm was superior to standard CG-algorithm with respect to the total of NOF , NOI and NOC.

We conclude that our new method is too effect when it compared with other established algorithms , In general , as well as the standard algorithm used so far showed superiority on problem of high dimensional and constrained.

**Table (1)Comparison of CG-algorithm with Self-Scaling algorithm**

| Test Fn. | CG-methods NOF(NOI)NOC | New(1) NOF(NOI)NOC | New(2) NOF(NOI)NOC | New(3) NOF(NOI)NOC |
|---|---|---|---|---|
| 1- | 661(222)431 | 660(224)434 | 642(220)429 | 654(220)431 |
| 2- | 12(4)52 | 10(6)38 | 10(6)38 | 166(80)97 |
| 3- | 83(33)66 | 55(23)32 | 51(21)26 | 51(21)26 |
| 4- | 44(19)21 | 59(24)36 | 34(18)20 | 32(3)819 |
| 5- | 585(40)46 | 161(30)38 | 66(16)20 | 110(32)67 |
| 6- | 55(21)25 | 46(21)23 | 81(18)21 | 45(17)22 |
| 7- | 96(21)9 | 76(26)7 | 94(16)9 | 56(22)6 |
| 8- | 38(17)24 | 38(18)23 | 41(19)33 | 80(35)112 |
| 9- | 43(21)29 | 64(24)146 | 53(19)134 | 52(18)149 |
| Tot. | 1617(398)703 | 1169(396)777 | 1072(353)730 | 1246(448)1729 |

## 8.Appendix:

The text functions in table (1) are [18]:

1-min $f(x) = (x_1 - 2)^2 + (x_1 - 2x_2)^2$

s.t $x_1^2 - x_2 = 0$

2. min $f(x) = x_1 x_2^2 + 2$

s.t $x_1^2 - x_2^2 = -2$

3-min $f(x) = (x_1 - 2)^2 + (x_1 - 2x_2)^2$

s.t $x_1^2 - x_2^2 = 4$

4-min $f(x) = x_1^2 + x_2^2$

s.t $x_1^2 - x_2^2 = -1$

5-min $f(x) = (x_1 - x_2)^2 + (x_2 - x_3)^4$

s.t $x_1 + x_1 x_2^2 + x_3^4 = 3$

6-min $f(x) = x_1^2 + x_2^2$

s.t $(x_1 - 1)^2 + x_2^2 = 0$

7-min

$$f(x) = 4x_1^2 + 2x_2^2 + 2x_3^2 - 33x_1 + 18x_2 - 24x_3$$

s.t
$$2x_2^2 - 3x_1 = 7$$
$$x_3^2 - 33x_1 = 11$$

8-min $f(x) = x_1^2 + x_2^2$

s.t $1 - x_1^2 - x_2^2 = 0$

9-min $f(x) = x_1^2 + x_2^2$

s.t $(x_1 - 1)^3 - x_2^2 + 4 = 0$

## Reference:


1. Kinsella J. , (2006), "Course Notes, for MS4327 Optimization " February 28.jkcray.math.ul.ie/msu327/slides. pdf.
2. McCormick, G. and Ritter, K. (1972)و ", Method of conjugate directions versus Quasi Newton methods", Math. Prog., 3.
3. Nazareth, L. (1979), "A relationship between the BFGS and conjugate gradient algorithms and its implications for new algorithms" SIAMJ. Numer. Anal. 16.794-800.
4. Bucley, A. and Lenir, A. (1983), " QN-like variable storage conjugate gradient", Math. Prog., 27,155-175.
5. Axelsson, O. (1974), " On preconditioning and convergence acceleration in sparse matrix problems;, CERN data handiling Divisions report.
6. Hager, W. W. and Zhang, H. (2005), " A new conjugate gradient methods with guaranteed descent and an efficient line search", SIAM Journal on Optimization, 16, 170-192.
7. Oren, S.S. and Luenberger, D.G (1974). "Self-scaling variable metric (SSVM) algorithms I: Criteria and sufficient conditions for scaling a class of algorithms," Management Science, 20, 845–862.
8. Byrd, R. Liu, D. and Nocedal, J. (1992). "On the behavior of Broyden's class of quasi- Newton methods," SIAM J. Optimization, 2, 533–557.

9. Al-Baali, M. (1998b), "Numerical experience with a class of self-scaling quasi-Newton algorithms," JOTA, 96, 533–553.
10. Andere'e, N, (2004), " A new gradient descent method for unconstrained optimization" ICI Technical Report.
11. Powell, M.J.D. (1986). "How bad are the BFGS and DFP methods when the objective function is quadratic," Math. Programming, 34, 34–47.
12. Zhang, Y. and Tewarson, R.P. (1988), "Quasi-Newton algorithms with updates from the preconvex part of Broyden's family," IMA J. Numer. Anal., 8, 487–509.
13. Nocedal, J.and Yuan, Y. (1993). "Analysis of a self-scaling quasi-Newton method," Math. Programming, 61, 19–37
14. Al-Baali, M. (1998a), "Analysis of a family of self-scaling quasi-Newton methods," Computational Optimization and Applications, 9, 191–203.
15. Oren, S.S. and Spedicato, E. (1976). "Optimal conditioning of self-scaling variable metric algorithms," Math. Programming, 10, 70–90.
16. Shanno, D.F. ,(1978), "*conjugate-Gradient Methods with inexact search*"Math. Of oper.Res 3.
17. Bunday, B.D.(1984), "*Basic Optimization Methods*", Edward Arnold, London.
18. Bazaraa, S. and Shetty C.M.,(1993)," non linear programming theory and algorithms" , New York , Chichester , Brisbane , Toronto.

# تطوير خوارزميات التدرج المترافق المشروط بطريقة لاكرانج

***ايمان طارق حامد**** ***هدى عصام احمد****

***أسيل مؤيد قاسم*****

*قسم البحوث والتقنيات الذكائية، كلية علوم الحاسبات والرياضيات، جامعة الموصل
** قسم الرياضيات، كلية التربية ،جامعة الموصل

**الخلاصة:**

اهتم البحث بتطوير طرائق التدرج المترافق ( CG) لحل مسائل الامثلية المقيدة غير الخطية باستخدام طريقة لاكرانج . معظم الخوارزميات العددية تكون حساسة للخطا الناتج من العمليات الحسابية لذلك اقترحنا طريقة تدرج مترافق ذاتية التقييس لتجاوز هذه الصعوبة و لزيادة قابلية الخوارزميات لحل المسائل العليلة وخوارزميتنا الجديدة المطورة لطريقة ال CG تبين ذلك وهي فعالة جدا عندما تقارن بالخوارزميات السابقة لحل مسائل الامثلية الثابتة القياسية.