ISSN 2278-3091
Volume 8, No.5, September - October 2019
International Journal of Advanced Trends in Computer Science and Engineering
Available Online at http://www.warse.org/IJATCSE/static/pdf/file/ijatcse80852019.pdf
https://doi.org/10.30534/ijatcse/2019/80852019

# On Modification of Preconditioning Conjugate Gradient Method with Self-Scaling Quasi-Newton

**Salah Ghaze Shareef[1], Hasan Hazim Jameel[2]**
[1]Department of Mathematics, Zakho University, Duhok, Kurdistan region-Iraq ,
Salah.gazi2014@gmail.com
[2]Department of Mathematics, Duhok University, Duhok, Kurdistan region- Iraq,
Hasan.hazim@uod.ac

## ABSTRACT

In this paper, a new class of self-scaling Quasi-Newton method updates for solving unconstrained non-linear optimization problem is investigated. The general strategy of self-scaling quasi newton is to scale hessian approximation matrix before it is updated at each iteration. This is to shun huge differences in the eigenvalues of the approximated Hessian of the objective function. The methods are convenient for large scale problems because the amount of storage required by the algorithms can be controlled by user. In comparison to standard serial Quasi-Newton methods, the suggested parallel self-scaling quasi newton algorithms show noticeable improvement in the total number of iterations and function/gradient evaluations needed for solving a broad extent of test problems.

**Key words**: Self-Scaling, PCG, SR1 method, quasi-Newton method, optimization.

## 1. INTRODUCTION

This study is launched into analyzing self-scaling SR1 method for solving the unconstrained minimization problem.

$$min\{f(x): x \in R^n\}, \qquad (1)$$

where f is a smooth function of n variables. At each $j^{th}$ , the iteration of the self-scaling method, a symmetric and positive definite matrix $H_j$ is given for the search direction to be computed by:

$$d_j = H_j\, g_j, \qquad (2)$$

where $g_j$ is the gradient evaluated at the current iterate $x_j$. One then computes the next iterate by

$$X_{j+1} = X_j + \delta_j d_j , \qquad (3)$$

Where $\delta_j$ is the step length and which obtained by Wolfe line search

$$f(x_j + \delta_j d_j) - f(x_j) \le \sigma_1 \delta_j g_j^T d_j \qquad (4)$$

$$g(x_j + \delta_j d_j) d_j \ge \sigma_2 g_j^T d_j \qquad (5)$$

Where $0 < \sigma_1 < \sigma_2 < 1$,which known as weak Wolfe condition [6,7]. Several self-scalings have been propounded by some scholars like Oren [9] who put forward some useful insights. With a self-scaling parameter k, this class of updates can be written as

$$H_{j+1} = \left(H_j - \frac{H_j y_j y_j^T H_j}{y_j^T H_j y_j} + \phi(y_j^T H_j y_j)\overline{v_j}\,\overline{v_j}^T\right)\omega_j + \frac{s_j^T s_j}{s_j^T s_j} \qquad (6)$$

Where

$s_j = x_{j+1} - x_j, \; y_j = \nabla f(x_{j+1}) - \nabla f(x_j), H_0 = I,$

$g_j = \nabla f(x_j)$

$$\bar{v} = \frac{s_j}{s_j^T y_j} + \frac{H_j y_j}{y_j^T H_j y_j}$$

Oren and Luenberger [8] suggest utilizing the self-adjusting values for the parameter

$$\omega_j = m\frac{g_j^T s_j}{g_j^T H_j y_j} + (1 - m)\frac{s_j^T y_j}{y_j^T H_j y_j}$$

And usually the value $t = 0$ is recommended for the update in the convex class.

To enhance the performance of Qusai- Newton update, Biggs proposes to opt $H_{j+1}$ to meet the following modified equation$H_{j+1} y_j = t_j s_j$ where $t_j > 0$ is a scaling parameter[2]. He showed that a modified BFGS could be derived as follows:

$$H_{(j+1)} = H_j + \frac{H_j y_j s_j^T + s_j y_j^T H_j}{s_j^T y_j} + \left(\frac{1}{\tau_j} + \frac{y_j^T H_j y_j}{s_j^T y_j}\right)\frac{s_j^T s_j}{s_j^T y_j} \qquad (7)$$

Where $\tau_j = \frac{1}{t_j} = \frac{6}{s_j^T y_j}\left(f(x_j) - f(x_{j+1}) + s_j^T g_{j+1}\right) - 2$

Also, Yang, Xu and Gao [10] made slight amendments for self-scaling symmetric rank one update with Davidon's optimal condition [3] (SHSR1) as follows:-

$$H_{j+1} = \rho_j H_j + \frac{(s_j - \rho_j H_j \hat{y})(s_j - \rho_j H_j \hat{y})^T}{\hat{y}_j^T (s_j - \rho_j H_j \hat{y})} \qquad (8)$$

Where $\rho_j$ is scaling factor,

$\hat{y} = \left(\frac{1+\theta_j}{s_j^T y_j}\right) y_j, \theta_j = 6(f_j - f_{j+1}) + 3(g_j + g_{j+1})^T s_j$ and $f_j = f(x_j)$.

## 2. SYMMETRIC RANK ONE MODIFICATION

Many approaches have been propounded to better the quasi-Newton Hessian approximation updates. In this section, we outline some recent suggested updates attained by modifying the vector $y_j$. Modifying $y_j$ was originally suggested by Powell who proposed a BFGS method for constrained optimization with

$$\bar{y} = y_j + (1-\theta)(G_j s_j - y_j) \quad (9)$$

where $0 < \theta < 1$, (see Al-Baali [4]).

In a bid to use the Hessian matrix in $H_j$ and Andrei [5] suggested a nonlinear conjugate gradient algorithm in which the Hessian/vector product $f(x_{j+1})v_j$ is approximated by finite differences:

$$\bar{y}_j = y_j + (1-\theta)\left(\frac{y_j}{\sigma} - y_j\right) \quad (10)$$

Where $\sigma = \frac{2\sqrt{\epsilon m}(1+\|w_{j+1}\|)}{\|s_j\|}$, and $\epsilon_m$ is error machine used for accuracy which is the smallest positive less than 1.

So, $H_{j+1}^{new} = H_j + \frac{(s_j - H_j\bar{y})(s_j - H_j\bar{y})^T}{\bar{y}_j^T (s_j - H_j\bar{y})}$ (11)

## 3. DERIVATION OF NEW SELF-SCALING QUASI-NEWTON METHOD SR1

In this section, we study the algorithm of PCG with new self- scaling Quasi-Newton methods SR1. The objective is to modify the performance of QN update, by selecting $H_{j+1}^{new}$ to satisfy Quasi-Newton condition:

$$H_{j+1}\bar{y}_j = \omega s_j \quad (12)$$

Consider

$d_{j+1} = -H_{j+1} g_{j+1}$

$d_{j+1}^T = -g_{j+1}^T H_{j+1}$ multiplying both sides from right by $\bar{y}_j$ we get

$$d_{j+1}^T \bar{y}_j = -g_{j+1}^T H_{j+1} \bar{y}_j \quad (13)$$

From (12) and (13) we get

$d_{j+1}^T \bar{y}_j = -g_{j+1}^T \omega_j s_j$

$d_{j+1}^T y_j \left(1 + (1-\theta)\left(\frac{1}{\sigma} - 1\right)\right) = -g_{j+1}^T \omega s_j$

It is defined $t$ scalar as below (see [Al-Assady [1] )

$d_{j+1}^T y_j = -\tau g_{j+1}^T s_j$

where $t = \frac{s_j^T y_j}{2 s_j^T g_j - 6(f_{j+1} - f_j)}$

$-\tau g_{j+1}^T s_j \left(1 + (1-\theta)\left(\frac{1}{\sigma} - 1\right)\right) = -g_{j+1}^T s_j \omega$

$\omega = \tau\left(1 + (1-\theta)\left(\frac{1}{\sigma} - 1\right)\right)$ where $0 < \theta < 1$, $\sigma$ is defined in (10).

Then, the new algorithm becomes as follows

$$H_{j+1} = H_j + \frac{(s_j - \omega H_j \bar{y}_j)(s_j - \omega H_j \bar{y}_j)^T}{\omega \bar{y}_j^T (s_j - \omega H_j \bar{y}_j)}, \quad \text{where } \omega > 0 \quad (14)$$

And the search direction of New algorithms is given by:

$$d_{j+1} = -H_{j+1} g_{j+1} + \frac{\bar{y}_j H_{j+1} g_{j+1}}{d_j^T \bar{y}_j} d_j \quad (15)$$

So, $H_{j+1}\bar{y}_j = \omega s_j$ is QN condition where $0 < \theta < 1$.

### 3.1 The Outlines of PCG-Method with New1.

Step 1. Let $x_0$ be initial point as well as identity $n x n$ symmetric positive definite matrix $H_0$, $\epsilon > 0$, $j = 0$.

Step 2. Compute $d_j = -H_j g_j$ where $g_j = \nabla f(x_j)$.

Step 3. Calculate $\delta_j$ to minimize $f(x_j + \delta_j d_j)$.

Step 4. Find new point of $x_{j+1} = x_j + \delta_j d_j$ and $y_j = g_{j+1} - g_j$.

Step5 Evaluate $g_{j+1} = \nabla f(x_{j+1})$, if $\|\nabla f(x_{j+1})\| < \epsilon$, then $x^* = x_{j+1}$ and stop.

Else find $s_j$ from $s_j = x_{j+1} - x_j$ go to step5.

Step 5 Evaluate $H_{j+1}$ by using (14).

Step 6 Evaluate $d_{j+1}$ from (15).

Step 7 set $j = j + 1$ go to step 2.

**Theorem 1.** If the new algorithm **New** is applied to the quadratic with Hessian matrix $G = G^T$, then $H_{j+1}\bar{y}_j = \omega s_j$, $j \geq 0$.

**Proof**. Multiplying both sides of (14) by $\bar{y}_j$ from right, we have

$$H_{j+1}\bar{y}_j = H_j \bar{y}_j + \frac{(\omega s_j - H_j \bar{y}_j)(\omega s_j - H_j \bar{y}_j)^T}{\bar{y}_j^T (\omega s_j - H_j \bar{y}_j)} \bar{y}_j \quad (16)$$

It is clear that $(\omega s_j - H_j \bar{y}_j)^T \bar{y}_j$ and $\bar{y}_j^T(\omega s_j - H_j \bar{y}_j)$ are scalars

So

$$(\omega s_j - H_j \bar{y}_j)^T \bar{y}_j = \bar{y}_j^T(\omega s_j - H_j \bar{y}_j) \quad (17)$$

Therefore

$$H_{j+1}\bar{y}_j = H_j \bar{y}_j + \omega s_j - H_j \bar{y}_j, \quad (18)$$

$H_{j+1}\bar{y}_j = \omega s_j$ ■

**Theorem 2.** If $H_j$ is a positive definite, then the matrix $H_{j+1}$ generated by the new algorithm is also positive definite.

**Proof.** Multiplying both sides of (14) by $\bar{y}_j$ from right and by $\bar{y}_j^T$ left, we have

$$\bar{y}_j^T H_{j+1} \bar{y}_j = \bar{y}_j^T H_j \bar{y}_j + \frac{\bar{y}_j^T (\omega s_j - H_j \bar{y}_j)(\omega s_j - H_j \bar{y}_j)^T \bar{y}_j}{\bar{y}_j^T (\omega s_j - H_j \bar{y}_j)} \quad (19)$$

So

$$\bar{y}_j^T H_{j+1} \bar{y}_j = \omega s_j^T \bar{y} \quad (20)$$

By substituting (10) in (20) we get

$\bar{y}_j^T H_{j+1} \bar{y}_j =$

$\omega s_j^T[(y_j + (1-\theta)\left(\frac{y_j}{\sigma} - y_j\right)] > 0$

$= \omega s_j^T y_j \left[1 + (1-\theta)\left(\frac{1}{\sigma} - 1\right)\right].$

Suppose that $k = \left[1 + (1-\theta)\left(\frac{1}{\sigma} - 1\right)\right]$,

If $k > 0$ and it is clear that $\omega > 0$

Since $s_j = \delta_j d_j$, $d_j = -H_j \nabla f(x_j)$

$s_j^T y_j = s_j^T \left(\nabla f(x_{j+1}) - \nabla f(x_j)\right)$

$= s_j^T \nabla f(x_{j+1}) - s_j^T \nabla f(x_j)$

$= \delta_j d_j^T \nabla f(x_{j+1}) + \delta_j \nabla f(x_j)^T H_j \nabla f(x_j)$

By using Wolfe condition (5)[6,7].

$s_j^T y_j \geq \delta_j \sigma_1 d_j^T \nabla f(x_j) + \delta_j \nabla f(x_j)^T H_j \nabla f(x_j)$

$= -\delta_j \sigma_1 \nabla f(x_j)^T H_j \nabla f(x_j) + \delta_j \nabla f(x_j)^T H_j \nabla f(x_j)$,

where $0 < \sigma_1 < 1$

$= (1-\sigma_1)\delta_j \nabla f(x_j)^T H_j \nabla f(x_j)$

Since $0 < \sigma_1 < 1$, and $H_j$ is positive definite and hence

$(1-\sigma_1)\delta_j \nabla f(x_j)^T H_j \nabla f(x_j) > 0$

$s_j^T y_j > 0$,

$s_j^T y_j \left[1 + (1-\theta)\left(\frac{1}{\sigma} - 1\right)\right] > 0.$

If $k < 0$. Since

$s_j^T y_j > s_j^T \nabla f(x_j)$

and

$-s_j^T \nabla f(x_j) = -\left(-\delta_j \nabla f(x_j)^T H_j \nabla f(x_j)\right)$

$= \delta_j \nabla f(x_j)^T H_j \nabla f(x_j)$

Therefore

$\omega s_j^T y_j > 0.$

$\bar{y}_j^T H_{j+1} \bar{y}_j > 0$■.

## 4. NUMERICAL RESULTS

This section is devoted to testing the implementation of the modified methods. Modified method is compared to the standard SR1. The results given in Table 1 specifically quote the NOI and NOF. The results in Table 1 illustrate that the modified SSSR1 method is superior to standard (SR1) method with respect to NOI and NOF.

**Table 1**: The comparison between modified and standard PCG Algorithm

| Test Function | N | Standard PCG | | New SSSR1 | |
|---|---|---|---|---|---|
| | | NOI | NOF | NOI | NOF |
| G-Central | 4 | 36 | 253 | 23 | 117 |
| | 100 | 43 | 331 | 26 | 153 |
| | 500 | 60 | 496 | 34 | 236 |
| | 1000 | 66 | 554 | 42 | 324 |
| | 5000 | 72 | 616 | 47 | 391 |
| Miele | 4 | 34 | 329 | 32 | 143 |
| | 100 | 47 | 182999 | 40 | 183 |
| | 500 | 53 | 183098 | 40 | 183 |
| | 1000 | 53 | 183098 | 47 | 219 |
| | 5000 | 65 | 189123 | 45 | 209 |
| Rosen | 4 | 31 | 90 | 30 | 87 |
| | 100 | 32 | 94 | 31 | 85 |
| | 500 | 33 | 98 | 31 | 85 |
| | 1000 | 37 | 115 | 34 | 92 |
| | 5000 | 37 | 120 | 37 | 99 |
| Cubic | 4 | 15 | 48 | 14 | 40 |
| | 100 | 16 | 66 | 14 | 41 |
| | 500 | 16 | 51 | 16 | 47 |
| | 1000 | 16 | 55 | 16 | 47 |
| | 5000 | 16 | 50 | 16 | 46 |
| G-Powell | 4 | 50 | 105 | 29 | 88 |
| | 100 | 72 | 228 | 21 | 61 |
| | 500 | 71 | 231 | 15 | 35 |
| | 1000 | 65 | 214 | 24 | 79 |
| | 5000 | 71 | 248 | 26 | 82 |
| Sum | 4 | 3 | 11 | 3 | 11 |
| | 100 | 14 | 83 | 14 | 81 |
| | 500 | 21 | 119 | 21 | 120 |
| | 1000 | 23 | 123 | 21 | 111 |
| | 5000 | 38 | 176 | 27 | 124 |
| Total | | 1206 | 743222 | 816 | 3619 |

**Table 2:** The rate of improvement between modified algorithm and Standard PCG.

| Tools | Standard PCG | SSSR1 |
|---|---|---|
| NOI | 100% | 67.6616 |
| NOF | 100% | 0.4869 |

The above table illustrates the rate of improvement in the modified algorithm self-scaling symmetric rank one with standard algorithm symmetric ran one. The numerical results of the new algorithm are better than the standard algorithm. As noted, the number of iterations and the number of functions of the standard algorithm are about 100%. That means that the new algorithm has improved as compared to standard algorithm with 32.3384 % in NOI and 99.5311% when $\theta \in (0,1)$.

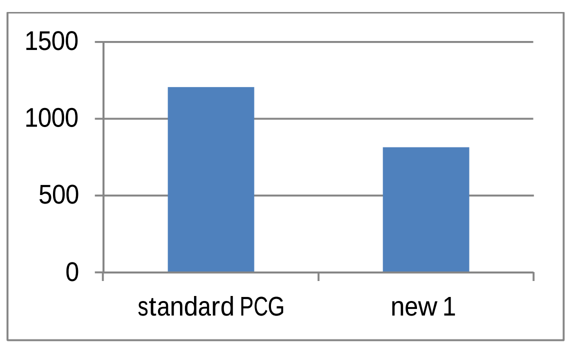


**Figure 1:** Comparison between standard PCG and the New algorithm according to NOI

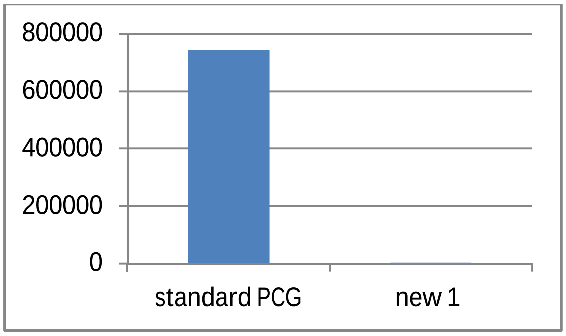


**Figure 2:** Comparison between standard PCG and the New algorithm according to NOF

## 5. CONCLUSION

This work, propounded a modification of self-scaling PCG (symmetric rank one) by using gradient-difference vector. The quasi-newton condition and positive definite have been proved. In addition to what is listed above of findings, the outcomes of a modified method self-scaling symmetric rank one are more superior and effective than the standard SR1.

## REFERENCES

1. Al-Assady, N. H. and huda, K. M. **A Rational logarithmic Science Model** for Unconstrained Non-linear Optimization, Rafideen Science J. 1997,Vol.8,pp.107-11.
2. Biggs, M. C. **A note on minimization algorithms which make use of non-quadratic properties of objective function**, J. Inst. Math Applies, 1973, Vol. 12, pp.337-338 https://doi.org/10.1093/imamat/12.3.337
3. L. P. Sun, **Updating of the self-scaling symmetric rank one algorithm with limited memory** for large-scale unconstrined optimization, Computational Optimization and Applications, 2004, 27. https://doi.org/10.1023/B:COAP.0000004977.04103.fb
4. M. Al-Baali and H. Khalfan, **an Overview of Some Practical QuasiNewton Methods** for Unconstrained Optimization, SQU Journal For Science, Sultan Qaboos University, 2007, 199-209. https://doi.org/10.24200/squjs.vol12iss2pp199-209
5. N. Andrei, 230, **Accelerated conjugate gradient algorithm with finite difference Hessian/vector product approximation** for unconstrained optimization, J. Comput. Appl. Math. 2. no. 2, 2009¸ 570–58 https://doi.org/10.1016/j.cam.2008.12.024
6. P. Wolfe**, Convergence** conditions for ascent methods II: Some corrections, SIAM Rev. 13 (1971),185_188. https://doi.org/10.1137/1013035
7. P. Wolfe, **Convergence conditions** for ascent methods, SIAM Rev. 11 (1969). 226_235. https://doi.org/10.1137/1011036
8. S. S. Oren and D. G. Luenberger, **self-scaling variable metric (SSVM) Algorithms**, **part I, criteria and Sufficient conditions** for scaling a class of Algorithms, Management science, 1974, vol. 20, pp. 845- 865. https://doi.org/10.1287/mnsc.20.5.845
9. S.S. Oren, **Self-scaling variable metric algorithms Part II**, Management Science, 1974, 20 863-874. https://doi.org/10.1287/mnsc.20.5.863
10. Y. T. Yang, C. X. Xu and Y. L. Gao, 34, **An optimal self-scaling strategy to the modified symmetric rank one updating**, Journal of Xian Jiaotong University, 2005, 100-103.