Yokohama Publishers
ISSN 1349-8169 ONLINE JOURNAL

Pacific Journal of Optimization
Vol. 7, No. 1, pp. 29–41, January 2011

POSITIVE-DEFINITE MEMORYLESS SYMMETRIC RANK ONE METHOD FOR LARGE-SCALE UNCONSTRAINED OPTIMIZATION*

Wah June Leong† and Malik Abu Hassan

Abstract: Memoryless quasi-Newton method is exactly the quasi-Newton method for which the approximation to the inverse of Hessian, at each step, is updated from a positive multiple of identity matrix. Hence its search direction can be computed without the storage of matrices, namely $O(n^2)$ storages. In this paper, a memoryless symmetric rank one (SR1) method for solving large-scale unconstrained optimization problems is presented. The basic idea is to incorporate the SR1 update within the framework of the memoryless quasi-Newton method. However, it is well-known that the SR1 update may not preserve positive definiteness even when updated from a positive definite matrix. Therefore, we propose that the memoryless SR1 method is updated from a positive scaled of the identity, in which the scaling factor is derived in such a way to preserve the positive definiteness and improves the condition of the scaled memoryless SR1 update. Under some standard conditions it is shown that the method is globally and $R-$linearly convergent. Numerical results show that the memoryless SR1 method is very encouraging.

Key words: *large-scale unconstrained optimization, symmetric rank one method, memoryless method, optimal scaling*

Mathematics Subject Classification: *65K10, 90C06, 90C52, 90C53*

---

1 Introduction

In this paper, the following unconstrained optimization problem is considered:

$$\min f(x); x \in R^n, \tag{1.1}$$

where $f : R^n \to R$ is assumed to be continuous differentiable function, and n, the dimension of the problem is large. Usually, problem (1.1) is solved iteratively through a line search scheme:

$$x_{k+1} = x_k + \lambda_k d_k, \tag{1.2}$$

where d_k is the search direction and $\lambda_k > 0$ is the steplength. The steplength can be calculated by an exact line search:

$$\lambda_k^* = \arg\min_{\lambda \in \Re}\{f(x_k + \lambda d_k)\}, \tag{1.3}$$

*This work is partially supported by the Malaysia MOHE FRGS Grant no. 5523383 and the Joint Chinese Academy of Sciences and Academy of Sciences for Developing Countries (CAS-TWAS) Fellowship no. 3240157252.

†Corresponding author

or by some line search conditions, such as Wolfe [19] conditions:

$$f(x_k + \lambda_k d_k) \leq f(x_k) + \beta_1 \lambda_k g_k^T d_k \tag{1.4}$$
$$g_{k+1}^T d_k \geq \beta_2 g_k^T d_k \tag{1.5}$$

where $0 < \beta_1 < 1/2$, $\beta_1 < \beta_2 < 1$ and $g_k = \bigtriangledown f(x_k)$ denotes the gradient vector of $f(x)$ at the current iteration point x_k.

We are particularly interested in elaborating an algorithm for solving very large cases, where the dimensions of the problems are up to 10^6. The need to solve these extremely large-scale optimization problems forces one to consider methods of $O(n)$ storage as the only methods of choice. This class of methods, includes those as the steepest descent method, conjugate gradient methods, limited memory quasi-Newton method and memoryless quasi-Newton method.

Memoryless quasi-Newton methods or one step limited memory quasi-Newton methods were first considered by Perry [15] and Shanno [17]. They are actually the quasi-Newton method for which at each iteration, a periodically restarted quasi-Newton correction is calculated from the initial approximation, commonly given by a positive multiple of identity matrix. Hence the memoryless quasi-Newton directions can be computed without the storage of matrices, namely $O(n^2)$ storages. Among the well-studied memoryless quasi-Newton methods is the memoryless BFGS method, which uses the BFGS update:

$$H_{k+1} = \left(I - \frac{y_k^T s_k}{s_k^T y_k}\right) H_k \left(I - \frac{y_k^T s_k}{s_k^T y_k}\right) + \frac{s_k s_k^T}{s_k^T y_k}, \tag{1.6}$$

where $s_k = x_{k+1} - x_k$ and $y_k = g_{k+1} - g_k$. In fact, a result by Shanno [17] shows that traditional CG methods such as the Fletcher-Reeves and Polak-Ribiére algorithm can be interpreted as a memoryless BFGS algorithm. Besides the BFGS update, one can extend the idea of memoryless updating to SR1 update:

$$H_{k+1} = H_k + \frac{(s_k - H_k y_k)(s_k - H_k y_k)^T}{y_k^T (s_k - H_k y_k)}. \tag{1.7}$$

and get the memoryless SR1 method. Minimization algorithms using SR1 update in both a line search and trust region context have been shown in computational experiments by Conn et al. [4] and Khalfan et al. [8] to be competitive with methods using the widely accepted BFGS update. Hence, it might be reasonable to think that such promising results can be extended to the memoryless version of SR1 method as well. However, it is well-known that the SR1 update may not preserve positive definiteness even when updated from a positive definite matrix. Therefore, to overcome this drawback, we propose a scaled memoryless SR1 method, which uses a periodically restarted SR1 correction from a positive scaled identity matrix. The scaling factor is derived in such a way the positive definiteness of the updated SR1 matrix can be preserved naturally and the condition of the SR1 update is also improved.

This paper is organized as follows: in Section 2, we discuss the optimal scaling factor for the identity matrix. Section 3 gives the convergence result of the scaled memoryless SR1 method for a convex function. Finally we include some numerical tests on a standard set of test problems in Section 4.

2 Optimal Scaling under the σ Measure

Throughout this section, we will assume that the curvature condition $y_k^T s_k > 0$. Let B_k be the current Hessian approximation, and its updated version B_{k+1} is computed by the direct

SR1 update:

$$B_{k+1} = B_k + \frac{(y_k - B_k s_k)(y_k - B_k s_k)^T}{s_k^T(y_k - B_k s_k)}. \tag{2.1}$$

Here when we mention inverse SR1 update, we mean the updating formula (1.7), otherwise the direct SR1 update is given by (2.1). Since memoryless quasi-Newton methods employ periodically restart at each iteration, commonly by a positive multiple of identity matrix, one can view the memoryless SR1 updating formula as the standard SR1 update (2.1) with $B_k = (1/\gamma)I$ for some $\gamma > 0$.

Hence, our primary aim is to find a scaling γ such that the direct SR1 formula, B_{k+1} updated from $(1/\gamma)I$ is 'optimal' under some measurements, while satisfying the secant equation and preserving positive definiteness for B_{k+1}. To date, various measures have been used to derive the optimal scaling factor for many well-known quasi-Newton updates. Commonly used is the $\kappa-$measure defined by

$$\kappa(A) = \frac{\xi_{\max}}{\xi_{\min}},$$

(the l_2-condition number) where A is an $n \times n$ positive definite matrix, $\xi_{\max}$ and $\xi_{\min}$ is the largest and smallest eigenvalue of A, respectively. This measure has been used by Davidon [6] to choose an optimally conditioned update in Broyden class and also by Shanno and Phua [18] to derive the optimal scaling factor for the BFGS update. However, since it is difficult to find the optimal scaling factor for SR1 update in l_2-condition number (see Wolkowicz [5] for details), one may consider the following measure, which is suggested by Dennis and Wolkowicz [5]:

$$\sigma(A) = \frac{\xi_{\max}}{\det(A)^{1/n}}, \tag{2.2}$$

($\sigma-$condition number) where det denotes determinant. Here, the measure σ acts as a condition number in that it provides a deviation from a multiple identity as does the l_2-conditioned number, κ. In fact, both Dennis and Wolkowicz [**5**] and Wolkowicz [20] had shown that any $\sigma-$optimal update will also be $\kappa-$optimal as well and have a common spectral property.

To motivate our memoryless update, we give the following result which is due to Leong and Hassan [9]:

Lemma 2.1. *Let*

$$\gamma_k = \frac{y_k^T y_k}{s_k^T y_k} - \left[\left(\frac{y_k^T y_k}{s_k^T y_k}\right)^2 - \frac{y_k^T y_k}{s_k^T s_k}\right]^{1/2}. \tag{2.3}$$

Then the direct SR1 matrix updated from $\frac{1}{\gamma_k}I$*:*

$$B_{k+1} = \frac{1}{\gamma_k}I + \frac{(y_k - (1/\gamma_k)s_k)\,(y_k - (1/\gamma_k)s_k))^T}{s_k^T\,(y_k - (1/\gamma_k)s_k)} \tag{2.4}$$

is the unique solution of

$$\begin{aligned} &\mathit{min} && \sigma(B_{k+1}^{-1}) \\ &\mathit{s.t.} && B_{k+1}^{-1} y_k = s_k \\ &\mathit{and} && B_{k+1}^{-1} \text{ is positive definite.} \end{aligned}$$

Note that, however, the denominator $s_k^T(y_k - (1/\gamma_k)s_k)$ in (2.4) may become zero and subsequently the matrix B_{k+1} generated by (2.4) is undefined. To deal with this difficulty, one can let $B_{k+1} = \frac{1}{\gamma_k}I$ whenever this difficulty arises. Hence, together with this safeguarding, we can give the following definition of B_{k+1}:

$$B_{k+1} = \begin{cases} \frac{1}{\gamma_k}I + \frac{(y_k-(1/\gamma_k)s_k)(y_k-(1/\gamma_k)s_k))^T}{s_k^T(y_k-(1/\gamma_k)s_k)} & ; \text{ if } s_k^T(y_k-(1/\gamma_k)s_k) \neq 0, \\ \frac{1}{\gamma_k}I & ; \text{ if } s_k^T(y_k-(1/\gamma_k)s_k) = 0. \end{cases} \tag{2.5}$$

Observe that in the latter case of (2.5), we will obtain $\gamma_k = \frac{s_k^T s_k}{s_k^T y_k}$. This value of γ_k is equal to the first stepsize formula proposed by Barzilai and Borwein [2], in which γ is chosen such that the matrix $B_{k+1} = (1/\gamma_k)I$ satisfies the following quasi-Newton property:

$$B_{k+1} = \arg \min_{B=(1/\gamma)I} \|Bs_k - y_k\|_2.$$

Next, by interchange the role of s and y, one can also obtain the following result:

Lemma 2.2. *Let*

$$\gamma_k = \frac{s_k^T s_k}{s_k^T y_k} - \left[\left(\frac{s_k^T s_k}{s_k^T y_k}\right)^2 - \frac{s_k^T s_k}{y_k^T y_k}\right]^{1/2}. \tag{2.6}$$

Then the inverse SR1 matrix updated from $\gamma_k I$:

$$H_{k+1} = \gamma_k I + \frac{(s_k - \gamma_k y_k)(s_k - \gamma_k y_k)^T}{y_k^T(s_k - \gamma_k y_k)} \tag{2.7}$$

is the unique solution of

$$\begin{aligned} &\textit{min} && \sigma(H_{k+1}^{-1}) \\ &\textit{s.t.} && H_{k+1}^{-1}s_k = y_k \\ &\textit{and} && H_{k+1}^{-1} \textit{ is positive definite.} \end{aligned}$$

Because of the same reason that is stated above, we use the following updating formula for H_{k+1}:

$$H_{k+1} = \begin{cases} \gamma_k I + \frac{(s_k-\gamma_k y_k)(s_k-\gamma_k y_k)^T}{y_k^T(s_k-\gamma_k y_k)} & ; \text{ if } y_k^T(s_k-\gamma_k y_k) \neq 0, \\ \gamma_k I & ; \text{ if } y_k^T(s_k-\gamma_k y_k) = 0. \end{cases} \tag{2.8}$$

Equivalently, the value of γ_k in the second case of (2.8) is equal to $\frac{y_k^T y_k}{s_k^T y_k}$, which is also the second stepsize formula of Barzilai-Borwein method. In addition, the corresponding $H_{k+1} = \gamma_k I$ is also satisfying the following:

$$H_{k+1} = \arg \min_{H=\gamma I} \|Hy_k - s_k\|_2.$$

For algorithmic purpose, we adopt formula (2.8) and compute our scaled memoryless SR1 direction, $d_k = -H_k g_k$ as follows:

1. If $y_k^T(s_k - \gamma_k y_k) \neq 0$:

$$\begin{aligned} d_k = -\gamma_{k-1} g_k \quad &+ \quad \gamma_{k-1} \left(\frac{s_{k-1}^T g_k - \gamma_{k-1} y_{k-1}^T g_k}{y_{k-1}^T s_{k-1} - \gamma_{k-1} y_{k-1}^T y_{k-1}} \right) y_{k-1} \\ &- \quad \left(\frac{s_{k-1}^T g_k - \gamma_{k-1} y_{k-1}^T g_k}{y_{k-1}^T s_{k-1} - \gamma_{k-1} y_{k-1}^T y_{k-1}} \right) s_{k-1}, \end{aligned} \tag{2.9}$$

where γ_{k-1} is given by (2.6) with the index k be replaced by $k-1$.

2. If $y_k^T (s_k - \gamma_k y_k) = 0$:

$$d_k = -\gamma_{k-1} g_k \tag{2.10}$$

where $\gamma_{k-1} = \frac{y_{k-1}^T y_{k-1}}{s_{k-1}^T y_{k-1}}$.

Finally, note that the computation of (2.9) involving only 4 vector products and requires only $3n$ storage requirements.

3 Convergence Results

For the analysis of this section, we make the following assumptions about the objective function f:

Assumption 3.1. Let G be the matrix of second derivatives of f.

1. The objective function f is twice continuously differentiable.
2. The level set $D = \{x \in R^n : f(x_0) \leq f(x)\}$ is convex.
3. There exist positive constants M_1 and M_2 such that

$$M_1 \|z\|^2 \leq z^T G(x) z \leq M_2 \|z\|^2 \tag{3.1}$$

for all $z \in R^n$ and all $x \in D$.

Before we proceed further, we give the following result on the boundedness of $\|B_k\|$:

Lemma 3.2. *Let x_0 be a starting point for which f satisfies Assumption 3.1. Then for any positive definite B_0, the sequence $\{\|B_k\|\}$ generated by (2.5) is bounded for all k if $s_k \neq 0$.*

Proof. Since $y_k = \bar{G}_k s_k$ where $\bar{G}_k = \int_0^1 G(x_k + \theta s_k) d\theta$, we have

$$s_k^T y_k = s_k^T \bar{G}_k s_k \quad \text{and} \quad y_k^T y_k = s_k^T \bar{G}_k^2 s_k, \tag{3.2}$$

which also implies that both $s_k^T y_k$ and $y_k^T y_k$ are bounded away from 0 under Assumption 1.3. Hence we can show the boundedness of $\frac{y_k^T y_k}{s_k^T y_k}$ and $\frac{y_k^T y_k}{s_k^T s_k}$ as follows:

$$M_1 \leq \frac{y_k^T y_k}{s_k^T y_k} \leq M_2 \quad \text{and} \quad M_1^2 \leq \frac{y_k^T y_k}{s_k^T s_k} \leq M_2^2 \tag{3.3}$$

(see Section 6.4 of Nocedal and Wright [14]).

Obviously, $\|B_{k+1}\|$ is bounded if B_{k+1} is defined by $(1/\gamma_k)I$ where $\gamma_k = \frac{s_k^T s_k}{s_k^T y_k}$. On the other hand, if B_{k+1} is defined by (2.4) where γ_k is given by (2.3), then by using both Theorem 3.1 and Corollary 3.1 of Wolkowicz [20], one can show that the distinct eigenvalues of B_{k+1} are:

$$\frac{1}{\gamma_k} \quad \text{and} \quad \frac{1}{\hat{\gamma_k}} \tag{3.4}$$

where $\hat{\gamma_k} = \frac{y_k^T y_k}{s_k^T y_k} + \left[\left(\frac{y_k^T y_k}{s_k^T y_k}\right)^2 - \frac{y_k^T y_k}{s_k^T s_k}\right]^{1/2}$. Furthermore, by utilizing the Cauchy-Schwarz inequality, we have

$$\left(\frac{y_k^T y_k}{s_k^T y_k}\right)^2 - \frac{y_k^T y_k}{s_k^T s_k} = \left(\frac{y_k^T y_k}{s_k^T y_k}\right)^2 \left(1 - \frac{(s_k^T y_k)^2}{(s_k^T s_k)(y_k^T y_k)}\right) > 0$$

and yields $\hat{\gamma_k} > \gamma_k$. In addition, since we can rewrite $\hat{\gamma_k}$ and γ_k as follows:

$$\hat{\gamma_k} = \frac{y_k^T y_k}{s_k^T y_k}\left[1 + \left(1 - \frac{s_k^T y_k / s_k^T s_k}{y_k^T y_k / s_k^T y_k}\right)^{1/2}\right],$$

and

$$\gamma_k = \frac{y_k^T y_k}{s_k^T y_k}\left[1 - \left(1 - \frac{s_k^T y_k / s_k^T s_k}{y_k^T y_k / s_k^T y_k}\right)^{1/2}\right],$$

it follows that

$$0 < M_1\left[1 - \left(1 - \frac{M_1}{M_2}\right)^{1/2}\right] \le \gamma_k < \hat{\gamma_k} \le M_2\left[1 + \left(1 - \frac{M_1}{M_2}\right)^{1/2}\right].$$

This implies that there exist positive constants q and Q where

$$q = \frac{1}{M_2\left[1 + \left(1 - \frac{M_1}{M_2}\right)^{1/2}\right]} \quad \text{and} \quad Q = \frac{1}{M_1\left[1 - \left(1 - \frac{M_1}{M_2}\right)^{1/2}\right]} \tag{3.5}$$

such that $q \le \mu_i \le Q$ for each eigenvalues μ_i of B_{k+1}. It follows that the sequence $\{\|B_k\|\}$ is also bounded, i.e.

$$q\|v\|^2 \le v^T B_k v \le Q\|v\|^2 \tag{3.6}$$

for all k and $v \in R^n$. □

Theorem 3.3. *Let x_0 be a starting point for which f satisfies Assumption 3.1. Consider $\{x_k\}$ the a sequence of points generated by the updating scheme $x_{k+1} = x_k - \lambda_k B_k^{-1} g_k$ where B_k is defined by (2.4) and λ_k satisfies the Wolfe conditions (1.4)-(1.5). Then the sequence $\{x_k\}$ converges globally to x^*. Moreover there is a constant $0 \le r < 1$ such that*

$$f(x_k) - f(x^*) \le r^k(f(x_0) - f(x^*)) \tag{3.7}$$

which implies that $\{x_k\}$ converges $R-$linearly.

Proof. Using Wolfe condition (1.4), the positive-definiteness and boundedness of the memoryless SR1 matrix, it follows that

$$f(x_{k+1}) \le f(x_k) - \beta_1 \lambda_k q \|g_k\|^2, \tag{3.8}$$

for some positive constants q. Therefore $f(x_{k+1}) \leq f(x_k)$ for all k and since f is bounded below, it follows that

$$\lim_{k\to\infty} f(x_k) - f(x_{k+1}) = 0.$$

As a consequence $\|g_k\|$ goes to zero, i.e. x_k converges to x^*.

Furthermore, since each eigenvalue μ_i of B_{k+1} is bounded by q and Q such that $q \leq \mu_i \leq Q$ for q and Q that are given by (3.5), we can see that the trace of B_{k+1} is bounded above:

$$tr(B_{k+1}) \leq nQ \tag{3.9}$$

and the determinant of B_{k+1} is bounded below:

$$det(B_{k+1}) \geq q^n. \tag{3.10}$$

(In the case where $B_{k+1} = (1/\gamma_k)I$ is used, we have $tr(B_{k+1}) \leq n/M_1$ and $det(B_{k+1}) \geq 1/M_2^n$.) Therefore from (3.9) and (3.10), we conclude that there exists a constant positive δ such that

$$\cos\theta_k = \frac{s_k^T B_k s_k}{\|s_k\|\|B_k s_k\|} \geq \delta, \quad \forall k.$$

One can show that the line search conditions (1.4)-(1.5) and Assumption 3.1 (see for example, Powell [16]) imply that there is a constant $c > 0$ such that

$$f(x_{k+1}) - f(x^*) \leq (1 - c\cos^2\theta_k)(f(x_k) - f(x^*)). \tag{3.11}$$

Applying (3.11) recursively we obtain (3.7). Finally, from (3.1)

$$\frac{1}{2}M_1\|x_k - x^*\|^2 \leq f(x_k) - f(x^*),$$

which together with (3.7) implies $\|x_k - x^*\| \leq r^{k/2}[2(f(x_0) - f(x^*))/M_1]^{1/2}$ so that the sequence $\{x_k\}$ is also $R-$linearly converged. □

4 Numerical Results

In this section we give some numerical results on solving a set of 36 general test problems with dimensions varying from 10^4 to 10^6. Table 1 presents names, and references of the problem set.

The algorithm, in general is given as follows:

Step 1. Consider an initial point x_0 and set $k = 0$.

Step 2. Compute the search direction d_k (let $d_0 = -g_0$).

Step 3. Find a value λ_k via the line search procedure. Update $x_{k+1} = x_k + \lambda_k d_k$.

Step 4. Test a criterion for stopping the iterations. If the test satisfied, then stop, else set $k := k + 1$ and return to Step 2.

For each algorithm, we use a line search routine of Moré and Thuente [13], which is based on cubic interpolation and satisfies the Wolfe conditions (1.4)-(1.5). The line search parameters are chosen as: $\beta_1 = 10^{-4}$, $\beta_2 = 0.9$. Default values are used for all other parameters, and the stopping criterion is that

$$\|g_k\| < 10^{-5} \tag{4.1}$$

is satisfied. We also force the algorithm to stop when the number of iterations excess 1000 and the number of function/gradient calls excess 10000. All codes are written in Fortran77 and in double precision arithmetic. All runs are performed on a PC with CoreDuo CPU. The methods tested include:

1. MLSR1: Memoryless SR1 method with the search direction given by (2.9).
2. MLBFGS: Memoryless BFGS method. It is exactly the limited memory BFGS method of Liu and Nocedal [10] with $m = 1$.
3. CG-FR: CG method which uses the Fletcher-Reeve formula with Powell's restart.
4. CG-PR: CG method which uses the Polak-Ribière formula with Powell's restart.
5. LBFGS(5): The limited memory BFGS method of Liu and Nocedal [10] with $m = 5$.
6. LBFGS(7): The limited memory BFGS method of Liu and Nocedal [10] with $m = 7$.

The performances of these algorithms, relative to number of iterations and number of function/gradient calls, are evaluated using the profiles of Dolan and Morè [7]. The numerical comparative results for $n = 10^4, 5 \times 10^4, 10^5$ are given in Figure 1-2. In addition, we also give in Table 2, the detail numerical results for the all six algorithms in solving problems with dimension 10^6. For this purpose, in Table 2 we give: n_I and $n_{f/g}$ denote the number of iterations and effective calls for function and gradient evaluation. The symbol – in the table indicates that either the method failed to initial or failed to converge within 999 iteration or the number of function/gradient evaluations exceeds 10000.

In this series of experiments, both MLSR1 and MLBFGS perform reasonably well when compared with those LBFGS and CG methods. However, it is shown that in general both LBFGS(5) and LBFGS(7) require somehow lesser function/gradient calls. While the LBFGS methods work well for moderate size problems, LBFGS(7) fails to start when attempts to solve problems of dimension 10^6 due to the "out-of-memory" situation. Furthermore, the figures also indicate that CG methods, in particular CG-FR seems to be the worst by comparison with the other algorithms. This is not surprising that without an efficient scaling/preconditioning strategy, especially when solving large-scale problems, CG methods are necessary inferior. Table 2 also shows that that memoryless quasi-Newton method is a good alternative if the dimensions of the problem are very large. Finally, we can conclude that the memoryless method could be a reliable method for large-scale optimization.

5 Conclusion

This paper proposed algorithm based on employing SR1 update within the memoryless quasi-Newton framework for solving large-scale unconstrained optimization. The proposed method uses a scaled identity matrix to update SR1 matrix, in which the scaling factor is derived in such a way that the scaled memoryless SR1 update is optimally conditioned and the lack of positive definiteness is eliminated. For a wider perspective, the memoryless SR1 method is appealing for several reasons: it is simple to implement, low storage requirement, globally converged and possesses $R-$linear rate of convergence.

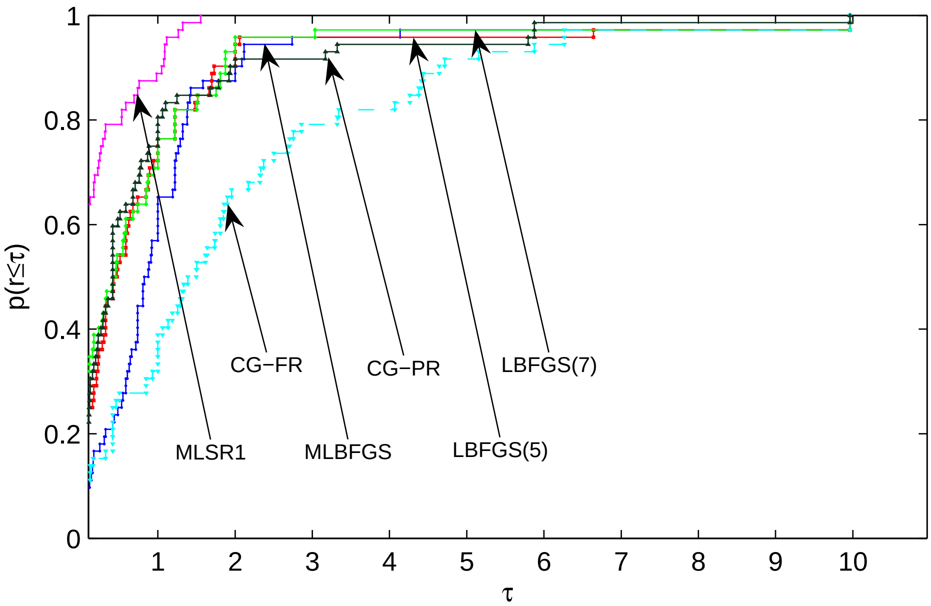

Figure 1: Performance profile based on iterations

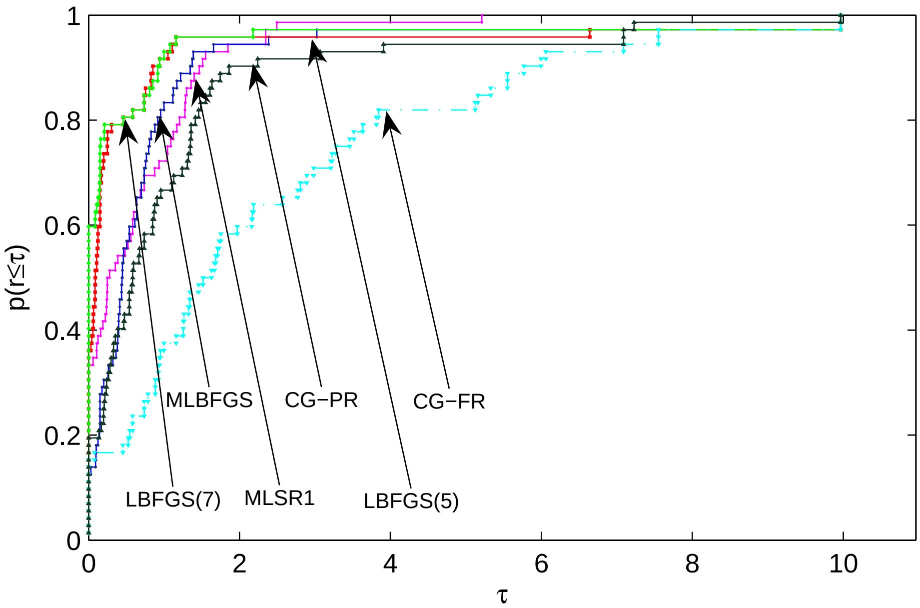

Figure 2: Performance profile based on function/gradient calls

Acknowledgements

The authors are thankful to the anonymous referees for their very valuable comments.

References

[1] N. Andrei, An unconstrained optimization test functions collection, *Adv. Model. Optim.* 10 (2008) 147–161.

[2] J. Barzilai and J.M. Borwein, Two-point step size gradient methods, *IMA J. Numer. Anal.* 8 (1988) 141–148.

[3] I. Bongartz, A.R. Conn, N.I.M. Gould and Ph. L. Toint, CUTE: constrained and unconstrained testing environment, *ACM Trans. in Mathematical Software* 21 (1995) 123–160.

[4] A.R. Conn, N.I.M. Gould and Ph. L. Toint, Convergence of quasi-Newton matrices generated by the symmetric rank one update, *Math. Program.* 48 (1991) 549–560.

[5] J.E. Dennis and H. Wolkowicz, Sizing and least change secant methods, *SIAM J. Numer. Anal.* 30 (1993) 1291–1313.

[6] W.C. Davidon, Optimally conditioned optimization algorithms without line search, *Math. Program.* 9 (1975) 1–30.

[7] E.D. Dolan and J.J. More, Benchmarking optimization software with perpormance profiles, *Math. Program.* 91 (2002) 201–213.

[8] H. Khalfan, R.H. Byrd and R.B. Schnabel, A theoretical and experimental study of the symmetric rank one update, *SIAM J. Optim.* 3 (1993) 1–24.

[9] W.J. Leong and M.A. Hassan, A restarting for the symmetric rank one update for unconstrained optimization. *Comput. Optim. Appl.* 42 (2009) 327–334.

[10] D. C. Liu and J. Nocedal, On the limited memory method for large-scale optimization, *Math. Program. B* 45 (1989) 503–528.

[11] A. W. Marshall and I. Olkin, *Inequalities: Theory of Majorization and its Applications.* Academic Press, New York, 1979.

[12] J.J. Moré, B.S. Garbow and K.E. Hillstrom, Line search algorithm with guaranteed sufficient decrease, *ACM Trans. in Mathematical Software* 7 (1981) 17–41.

[13] J.J. Moré and D.J. Thuente, Testing unconstrained optimization software, *ACM Trans. in Mathematical Software* 20 (1994) 286–307.

[14] J. Nocedal and S.J. Wright, *Numerical Optimization: Series in Operation Research*, Springer, Berlin, 1999.

[15] J. M. Perry, *A class of conjugate gradient algorithm with a two step variable metric memory*, Discussion Paper 269, Center for Mathematical Studies in Economics and Management Science, Northwestern Univ., Chicago, 1977.

[16] M.J.D. Powell, *Some global convergence properties of avariable metric algorithm for minimization without exact linesearch* , R. W Cottle and C. E. Lemke eds. Nonlinear Programing, SIAM-AMS Proceedings IX, SIAM Publications, Philadelphia, 1976.

[17] D.F. Shanno, Conjugate gradient methods with inexact searches, *Math. Oper. Res.* 3 (1978) 244–256.

[18] D.F. Shanno and K.H. Phua, Matrix conditioning and nonlinear optimization, *Math. Program* 14 (1978) 149–160.

[19] P. Wolfe, Convergence conditions for ascent methods, *SIAM Review* 11 (1968) 226–235.

[20] H. Wolkowicz, Measure for symmetric rank-one updates, *Math. Oper. Res.* 19 (1994) 815–830.

Manuscript received 22 January 2009
revised 28 October 2009, 15 January 2010
accepted for publication 18 February 2010

Wah June Leong
Institute for Mathematical Research, University Putra Malaysia
Serdang, 43400 Selangor, Malaysia and;
Institute of Computational Mathematics and Scientific/Engineering Computing
Academy of Mathematics and Systems Science
Chinese Academy of Sciences
E-mail address: `leong@math.upm.edu.my`

Malik Abu Hassan
Institute for Mathematical Research, University Putra Malaysia
Serdang, 43400 Selangor, Malaysia
E-mail address: `malik@fsas.upm.edu.my`

Table 1: List of test functions and their references

Function's name	Reference
Trigonometric	Moré et al. [12]
Extended Rosenbrock	Moré et al. [12]
Beale	Moré et al. [12]
Wood	Moré et al. [12]
Penalty I	Moré et al. [12]
Broyden Tridiagonal	Moré et al. [12]
Raydan	Andrei [1]
Extended White and Holst	Andrei [1]
Extended Tridiagonal	Andrei [1]
Extended Three Expo Term	Andrei [1]
Generalized Tridiagonal	Andrei [1]
Diagonal 4	Andrei [1]
Diagonal 5	Andrei [1]
Extended Maratos	Andrei [1]
Extended Block-Diagonal BD1	Andrei [1]
Extended Hiebert	Andrei [1]
Extended Quadratic Penalty QP2	Andrei [1]
Extended EP1	Andrei [1]
Extended Tridiagonal 2	Andrei [1]
Diagonal 6	Andrei [1]
ARWHEAD	CUTE [3]
NONDIA	CUTE [3]
DQDRTIC	CUTE [3]
DIXMAANA	CUTE [3]
DIXMAANB	CUTE [3]
DIXMAANC	CUTE [3]
HIMMELBC	CUTE [3]
CLIFF	CUTE [3]
EDENSCH	CUTE [3]
LIARWHD	CUTE [3]
ENGVAL1	CUTE [3]
FLETCHCR	CUTE [3]
COSINE	CUTE [3]
DENSCHNB	CUTE [3]
DENSCHNF	CUTE [3]
FREUROTH	CUTE [3]

Table 2: Results for the methods in solving problems with $n = 10^6$

Test function	MLSR1 $n_I/n_{f/g}$	MLBFGS $n_I/n_{f/g}$	CG-FR $n_I/n_{f/g}$	CG-PR $n_I/n_{f/g}$	LBFGS(5) $n_I/n_{f/g}$	LBFGS(7) $n_I/n_{f/g}$
Trigonometric	133/644	–	–	–	–	–
Ext Rosenbrock	25/59	43/69	182/315	28/58	37/53	–
Beale	12/23	19/30	18/32	17/35	14/18	–
Wood	113/207	98/166	118/229	62/136	69/100	–
Penalty I	–	–	–	–	–	–
Broyden Tridiagonal	67/106	37/48	–	88/139	45/51	–
Raydan	3/9	7/11	4/9	4/9	7/11	–
Ext White and Holst	25/47	56/90	134/246	34/69	39/52	–
Ext Tridiagonal	17/18	30/44	21/42	22/40	28/35	–
Ext Three Expo Term	–	–	35/122	12/24	–	–
Generalized Tridiagonal	–	–	–	–	–	–
Diagonal 4	–	–	13/27	7/13	–	–
Diagonal 5	–	–	6/33	6/33	–	–
Ext Maratos	69/70	88/144	–	101/420	88/144	–
Ext Block-Diagonal BD1	22/33	25/39	42/76	32/111	14/23	–
Ext Hiebert	52/114	98/16	118/229	62/136	69/100	–
Ext Quad Penalty QP2	28/81	50/81	291/444	31/80	57/83	–
Ext EP1	2/3	4/6	2/5	2/5	4/6	–
Ext Tridiagonal 2	17/28	30/44	21/42	14/40	28/35	–
Diagonal 6	3/9	7/11	4/9	4/9	7/11	–
ARWHEAD	28/37	10/16	8/95	15/169	13/18	–
NONDIA	3/7	4/5	4/7	4/7	4/5	–
DQDRTIC	30/60	30/41	120/191	40/77	11/19	–
DIXMAANA	11/16	11/15	13/26	9/17	12/16	–
DIXMAANB	10/11	11/15	12/21	12/21	11/15	–
DIXMAANC	13/22	13/17	15/30	15/29	13/17	–
HIMMELBC	6/15	19/26	14/26	9/18	8/15	–
CLIFF	21/98	50/108	55/112	29/51	53/58	–
EDENSCH	45/46	21/26	–	46/658	18/23	–
LIARWHD	17/35	39/61	81/172	21/43	33/40	–
ENGVAL1	–	–	–	–	–	–
FLETCHCR	–	–	–	–	–	–
COSINE	–	703/767	–	–	801/821	–
DENSCHNB	–	–	–	–	–	–
DENSCHNF	–	–	–	–	–	–
FREUROTH	11/24	36/51	129/618	60/169	17/22	–