Hindawi Publishing Corporation
Journal of Applied Mathematics
Volume 2012, Article ID 875494, 11 pages
doi:10.1155/2012/875494

Research Article

Accumulative Approach in Multistep Diagonal Gradient-Type Method for Large-Scale Unconstrained Optimization

Mahboubeh Farid,[1] Wah June Leong,[1] and Lihong Zheng[2]

[1] *Department of Mathematics, University Putra Malaysia, Selangor, 43400 Serdang, Malaysia*
[2] *School of Computing and Maths, Charles Sturt University, Mitchell, NSW 2795, Australia*

Correspondence should be addressed to Mahboubeh Farid, mfarid7@gmail.com

Received 2 April 2012; Revised 10 May 2012; Accepted 16 May 2012

Academic Editor: Vu Phat

This paper focuses on developing diagonal gradient-type methods that employ accumulative approach in multistep diagonal updating to determine a better Hessian approximation in each step. The interpolating curve is used to derive a generalization of the weak secant equation, which will carry the information of the local Hessian. The new parameterization of the interpolating curve in variable space is obtained by utilizing accumulative approach via a norm weighting defined by two positive definite weighting matrices. We also note that the storage needed for all computation of the proposed method is just $O(n)$. Numerical results show that the proposed algorithm is efficient and superior by comparison with some other gradient-type methods.

1. Introduction

Consider the unconstrained optimization problem:

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

where $f : R^n \to R$ is twice continuously differentiable function. The gradient-type methods for solving (1.1) can be written as

$$x_{k+1} = x_k - B_k^{-1} g_k, \tag{1.2}$$

where g_k and B_k denote the gradient and the Hessian approximation of f at x_k, respectively. By considering $B_k = \alpha_k I$, Barzilai and Borwein (BB) [1] give

$$\alpha_{k+1} = \frac{s_k^T y_k}{s_k^T s_k}, \tag{1.3}$$

where it is derived by minimizing $\|\alpha_{k+1} s_k - y_k\|_2$ respect to α with $s_k = x_{k+1} - x_k$ and $y_k = g_{k+1} - g_k$. Recently, some improved one-step gradient-type methods [2–5] in the frame of BB algorithm were proposed to solve (1.1). It is proposed to let B_k be a diagonal nonsingular approximation to the Hessian and a new approximating matrix B_{k+1} to the Hessian is developed based on weak secant equation of Dennis and Wolkowicz [6]

$$s_k^T B_{k+1} s_k = s_k^T y_k. \tag{1.4}$$

In one-step method, data from one previous step is used to revise the current approximation of Hessian. Later Farid and Leong [7, 8] proposed multistep diagonal gradient methods inspired by the multistep quasi-Newton method of Ford [9, 10]. In this multistep framework, a fixed-point approach for interpolating polynomials was derived from data in previous iterations (not only one previous step) [7–10]. General approach of multistep method is based on the measurement of distances in the variable space where the distance of every iterate is measured from one-selected iterate. In this paper, we are interested to develop multistep diagonal updating based on accumulative approach for defining new parameter value of interpolating curve. From this point, the distance is accumulated between consecutive iterates as they are traversed in the natural sequence. For measuring the distance, we need to parameterize the interpolating polynomial through a norm that is defined by a positive definite weighting matrix, say M. Therefore, the performance of the multistep method may be significantly improved by carefully defining the weighting matrix. The rest of paper is organized as follows. In Section 2, we discuss a new multistep diagonal updating scheme based on the accumulative approach. In Section 3, we establish the global convergence of our proposed method. Section 4 presents numerical result and comparisons with BB method and one-step diagonal gradient method are reported. Conclusions are given in Section 5.

2. Derivation of the New Diagonal Updating via Accumulative Approach

This section motivates to state new implicit updates for diagonal gradient-type method through accumulative approach to determining a better Hessian approximation at each iteration. In multistep diagonal updating methods, weak secant equation (1.4) may be generalized by means of interpolating polynomials, instead of employing data just from one previous iteration like in one-step methods. Our aim is to derive efficient strategies for choosing a suitable set of parameters to construct the interpolating curve and investigate the best norm for measurement of the distances required to parameterize the interpolating polynomials. In general, this method obeys the recursive formula of the form

$$x_{k+1} = x_k - \alpha_k B_k^{-1} g_k, \tag{2.1}$$

where x_k is the kth iteration point, α_k is step length which is determined by a line search, B_k is an approximation to the Hessian in a diagonal form, and g_k is the gradient of f at x_k. Consider a differentiable curve $x(\tau)$ in R^n. The derivative of $g(x(\tau))$, at point $x(\tau^*)$, can be obtained by applying the chain rule:

$$\left.\frac{dg}{d\tau}\right|_{\tau=\tau^*} = G(x(\tau))\left.\frac{dx}{d\tau}\right|_{\tau=\tau^*}. \tag{2.2}$$

We are interested to derive a relation that will be satisfied by the approximation of Hessian in diagonal form at x_{k+1}. If we assume that $x(\tau)$ passes through x_{k+1} and choose τ^* so that

$$x(\tau^*) = x_{k+1}, \tag{2.3}$$

then we have

$$G(x_{k+1})\frac{dx_{k+1}}{d\tau} = \frac{dg(x_{k+1})}{d\tau}. \tag{2.4}$$

As in this paper, we use two-step method, therefore; we use information of most recent points x_{k-1}, x_k, x_{k+1} and their associated gradients. Consider $x(\tau)$ as the interpolating vector polynomial of degree 2:

$$x(\tau_j) = x_{k+j-1} \quad j = 0, 1, 2. \tag{2.5}$$

The selection of distinct scalar value τ_j efficiently through the new approach is the main contribution of this paper and will be discussed later in this section. Let $h(\tau)$ be the interpolation for approximating the gradient vector:

$$h(\tau_j) = g(x_{k+j-1}) \quad j = 0, 1, 2. \tag{2.6}$$

By denoting $x(\tau_2) = x_{k+1}$ and defining

$$\begin{aligned} \frac{dx(\tau_2)}{d\tau} &= r_k, \\ \frac{dg(x(\tau_2))}{d\tau} &= w_k, \end{aligned} \tag{2.7}$$

we can obtain our desired relation that will be satisfied by the Hessian approximation at x_{k+1} in diagonal form. Corresponding to this two-step approach, weak secant equation will be generalized as follows:

$$r_k^T B_{k+1} r_k = r_k^T w_k. \tag{2.8}$$

Then, B_{k+1} can be obtained by using an appropriately modified version of diagonal updating formula in [3] as follows:

$$B_{k+1} = B_k + \frac{\left(r_k^T w_k - r_k^T B_k r_k\right)}{\operatorname{tr}\left(F_k^2\right)} F_k, \tag{2.9}$$

where $F_k = \operatorname{diag}((r_k^{(1)})^2, (r_k^{(2)})^2, \ldots, (r_k^{(n)})^2)$. Now, we attempt to construct an algorithm for finding desired vector r_k and w_k to improve the Hessian approximation. The proposed method is outlined as follows. First, we seek to derive strategies for choosing a suitable set of values τ_0, τ_1, and τ_2. The choice of $\{\tau_j\}_{j=0}^2$ is such that to reflect distances between iterates x_k in R^n that are dependent on some metric of the following general form:

$$\phi_M(z_1, z_2) = \left\{(z_1 - z_2)^T M (z_1 - z_2)\right\}^{1/2}. \tag{2.10}$$

The establishment on τ_j can be made via the so-called accumulative approach where the accumulating distances (measured by the metric ϕ_M) between consecutive iterates are used to approximate τ_j. This leads to the following definitions (where without loss of generality, we take τ_1 to be origin for value of τ):

$$\begin{aligned} \tau_1 &= 0, \\ \tau_j &= \tau_{j+1} - \phi_M\left(x_{k+j}, x_{k+j-1}\right) \qquad j = 0, 2. \end{aligned} \tag{2.11}$$

Then, we can construct the set $\{\tau_j\}_{j=0}^2$ as follows:

$$\begin{aligned} \tau_0 &= \tau_1 - \phi_M(x_k, x_{k-1}) \\ &\overset{\text{def}}{=} -\|x_k - x_{k-1}\|_M \\ &= -\|s_{k-1}\|_M, \\ \tau_2 &= \phi_M(x_{k+1}, x_k) \\ &\overset{\text{def}}{=} \|x_{k+1} - x_k\|_M \\ &= \|s_k\|_M, \end{aligned} \tag{2.12}$$

where r_k and w_k are depending on the value of τ. As the set $\{\tau_j\}_{j=0}^2$ measures the distances, therefore they need to be parameterized the interpolating polynomials via a norm defined by a positive definite matrix M. It is necessary to choose M with some care, while improving the approximation of Hessian can be strongly influenced via the choice of M. Two choices for the weighting matrix M are considered in this paper. In first choice, if $M = I$, the $\|\cdot\|_M$ reduces to the Euclidean norm, and then we obtain the following τ_j values accordingly:

$$\tau_2 = \|s_k\|_2, \qquad \tau_1 = 0, \qquad \tau_0 = -\|s_{k-1}\|_2. \tag{2.13}$$

The second choice of weighting matrix M is to take $M = B_k$, where the current B_k is diagonal approximation to the Hessian. By these two means, the measurement of the relevant distances is determined by the properties of the current quadratic approximation (based on B_k) to the objective function:

$$\tau_2 = (s_k B_k s_k)^{1/2}, \qquad \tau_1 = 0, \qquad \tau_0 = -(s_{k-1} B_k s_{k-1})^{1/2}. \tag{2.14}$$

Since B_k is a diagonal matrix, then it is not expensive to compute $\{\tau_j\}_{j=0}^{2}$ at each iteration. The quantity δ is introduced here and defined as follows:

$$\delta \stackrel{\text{def}}{=} \frac{\tau_2 - \tau_1}{\tau_1 - \tau_0}, \tag{2.15}$$

and r_k and w_k are given by the following expressions:

$$r_k = s_k - \frac{\delta^2}{1 + 2\delta} s_{k-1}, \tag{2.16}$$

$$w_k = y_k - \frac{\delta^2}{1 + 2\delta} y_{k-1}. \tag{2.17}$$

To safeguard on the possibility of having very small or very large $r_k^T w_k$, we require that the condition

$$\varepsilon_1 \|r_k\|_2^2 \leq r_k^T w_k \leq \varepsilon_2 \|r_k\|_2^2 \tag{2.18}$$

is satisfied (we use $\varepsilon_1 = 10^{-6}$ and $\varepsilon_2 = 10^6$). If not, then we replace $r_k = s_k$ and $w_k = y_k$. More that the Hessian approximation (B_{k+1}) might not preserve the positive definiteness in each step. One of the fundamental concepts in this paper is to determine an "improved" version of the Hessian approximation to be used even in computing the metric when $M = B_k$ and a weighing matrix as norm should be positive definite. To ensure that the updates remain positive definite, a scaling strategy proposed in [7] is applied. Hence, the new updating formula that incorporates the scaling strategy is given by

$$B_{k+1} = \eta_k B_k + \frac{\left[r_k^T w_k - \eta_k r_k^T B_k r_k\right]}{\operatorname{tr}\left(F_k^2\right)} F_k, \tag{2.19}$$

where

$$\eta_k = \min\left(\frac{r_k^T w_k}{r_k^T B_k r_k}, 1\right). \tag{2.20}$$

This guarantees that the updated Hessian approximation is positive. Finally, the new accumulative MD algorithm is outlined as follows.

2.1. Accumulative MD Algorithm

Step 1. Choose an initial point $x_0 \in R^n$, and a positive definite matrix $B_0 = I$.
Let $k := 0$.

Step 2. Compute g_k. If $\|g_k\| \leq \epsilon$, stop.

Step 3. If $k = 0$, set $x_1 = x_0 - (g_0/\|g_0\|)$. If $k = 1$ set $r_k = s_k$ and $w_k = y_k$ go to Step 5.

Step 4. If $k \geq 2$ and $M = I$ is considered, compute $\{\tau_j\}_{j=0}^{2}$ from (2.13).
Else if $M = B_k$, compute $\{\tau_j\}_{j=0}^{2}$ from (2.14).
Compute δ_k, r_k, w_k and η_k, from (2.15), (2.16), (2.17), and (2.20), respectively.
If $r_k^T w_k \leq 10^{-4}\|r_k\|_2\|w_k\|_2$, set $r_k = s_k$ and $w_k = y_k$.

Step 5. Compute $d_k = -B_k^{-1}g_k$ and calculate $\alpha_k > 0$ such that the Armijo [11], condition holds: $f(x_{k+1}) \leq f(x_k) + \sigma\alpha_k g_k^T d_k$, where $\sigma \in (0,1)$ is a given constant.

Step 6. Let $x_{k+1} = x_k - \alpha_k B_k^{-1}g_k$, and update B_{k+1} by (2.19).

Step 7. Set $k := k + 1$, and return to Step 2.

3. Convergence Analysis

This section is devoted to study the convergence of accumulative MD algorithm, when applied to the minimization of a convex function. To begin, we give the following result, which is due to Byrd and Nocedal [12] for the step generated by the Armijo line search algorithm. Here and elsewhere, $\|\cdot\|$ denotes the Euclidean norm.

Theorem 3.1. *Assume that f is a strictly convex function. Suppose the Armijo line search algorithm is employed in a way that for any d_k with $d_k^T g_k < 0$, the step length, α_k satisfies*

$$f(x_k + \alpha_k d_k) \leq f(x_k) + \sigma\alpha_k g_k^T d_k, \tag{3.1}$$

where $\alpha_k \in [\tau, \tau']$, $0 < \tau < \tau'$ and $\sigma \in (0,1)$. Then, there exist positive constants ρ_1 and ρ_2 such that either

$$f(x_k + \alpha_k d_k) - f(x_k) \leq -\rho_1 \frac{\left(d_k^T g_k\right)^2}{\|d_k\|^2} \tag{3.2}$$

or

$$f(x_k + \alpha_k d_k) - f(x_k) \leq -\rho_2 d_k^T g_k \tag{3.3}$$

is satisfied.

We can apply Theorem 3.1 to establish the convergence of some Armijo-type line search methods.

Theorem 3.2. *Assume that f is a strictly convex function. Suppose that the Armijo line search algorithm in Theorem 3.1 is employed with d_k chosen to obey the following conditions: there exist positive constants c_1 and c_2 such that*

$$-g_k^T d_k \geq c_1 \|g_k\|^2, \qquad \|d_k\| \leq c_2 \|g_k\|, \tag{3.4}$$

for all sufficiently large k. Then, the iterates x_k generated by the line search algorithm have the property that

$$\liminf_{k \to \infty} \|g_k\| = 0. \tag{3.5}$$

Proof. By (3.4), we have that either (3.2) or (3.6) becomes

$$f(x_k + \alpha_k d_k) - f(x_k) \leq -c \|g_k\|^2, \tag{3.6}$$

for some positive constants. Since f is strictly convex, it is also bounded below. Then, (3.1) implies that $f(x_k + \alpha_k d_k) - f(x_k) \to 0$ as $k \to \infty$. This also implies that $\|g_k\| \to 0$ as $k \to \infty$ or at least

$$\liminf_{k \to \infty} \|g_k\| = 0. \tag{3.7}$$

□

To prove that the accumulative MD algorithm is globally convergent when applied to the minimization of a convex function, it is sufficient to show that the sequence $\{\|B_k\|\}$ generated by (2.19)-(2.20) is bounded both above and below, for all finite k so that its associated search direction satisfies condition (3.4). Since B_k is diagonal, it is enough to show that each element of B_k says $B_k^{(i)}$; $i = 1, \ldots, n$ is bounded above and below by some positive constants. The following theorem gives the boundedness of $\{\|B_k\|\}$.

Theorem 3.3. *Assume that f is strictly convex function where there exists positive constants m and M such that*

$$m\|z\|^2 \leq z^T \nabla^2 f(x) z \leq M\|z\|^2, \tag{3.8}$$

for all $x, z \in R^n$. Let $\{\|B_k\|\}$ be a sequence generated by the accumulative MD method. Then, $\|B_k\|$ is bounded above and below for all finite k, by some positive constants.

Proof. Let $B_k^{(i)}$ be the ith element of B_k. Suppose B_0 is chosen such that $\omega_1 \leq B_0^{(i)} \leq \omega_2$; $i = 1, \ldots, n$, where ω_1 and ω_2 are some positive constants.

Case 1. If (2.18) is satisfied, we have

$$B_1 = \begin{cases} \eta_0 B_0; & \text{if } r_0^T w_0 < r_0^T B_0 r_0 \\ B_0 + \dfrac{r_0^T w_0 - r_0^T B_0 r_k}{\operatorname{tr}(F_0^2)} F_0; & \text{if } r_0^T w_0 \geq r_0^T B_0 r_0. \end{cases} \tag{3.9}$$

By (2.18) and the definition of η_k, one can obtain

$$\frac{\varepsilon_1}{\omega_2} \leq \eta_0 \leq 1. \tag{3.10}$$

Thus, if $r_0^T w_0 < r_0^T B_0 r_0$, then $B_1 = \eta_0 B_0$ satisfies

$$\frac{\varepsilon_1 \omega_1}{\omega_2} \leq B_1^{(i)} \leq \omega_2. \tag{3.11}$$

On the other hand, if $r_0^T w_0 \geq r_0^T B_0 r_0$, then

$$B_1^{(i)} = B_0^{(i)} + \frac{(r_0^T w_0 - r_0^T B_0 r_0)}{\operatorname{tr}(F_0^2)} \left(r_0^{(i)}\right)^2, \tag{3.12}$$

where $r_0^{(i)}$ is the ith component of r_0. Letting $(r_0^{(M)})$ be the largest component (in magnitude) of r_0, that is, $(r_0^{(i)})^2 \leq (r_0^{(M)})^2$; for all i, then it follows that $\|r_0\|^2 \leq n(r_0^{(M)})^2$, and the property of $r_0^T w_0 \geq r_0^T B_0 r_0$, (3.12) becomes

$$\omega_1 \leq B_0^{(i)} \leq B_1^{(i)} \leq \omega_2 + \frac{n(\varepsilon_2 - \omega_1)}{\operatorname{tr}(F_0^2)} \left(r_0^{(M)}\right)^4 \leq \omega_2 + n(\varepsilon_2 - \omega_1). \tag{3.13}$$

Hence, $B_1^{(i)}$ is bounded above and below, for all i in both occasions.

Case 2. If (2.18) is violated, then the updating formula for B_1 becomes

$$B_1^{(i)} = \eta_0 B_0^{(i)} + \frac{(s_0^T y_0 - \eta_0 s_0^T B_0 s_0)}{\operatorname{tr}(E_0^2)} \left(s_0^{(i)}\right)^2, \tag{3.14}$$

where $s_0^{(i)}$ is the ith component of s_0, $E_0 = \operatorname{diag}((s_0^{(1)})^2, (s_0^{(2)})^2, \ldots, (s_0^{(n)})^2)$, and $\eta_0 = \min(1, s_0^T y_0 / s_0^T B_0 s_0)$.

Because $\eta_0 \leq 1$ also implies that $s_0^T y_0 - s_0^T B_0 s_0 \geq 0$, then this fact, together with the convexity property (3.8), and the definition of η give

$$\min\left(1, \frac{m}{\omega_2}\right)\omega_1 \leq \eta_0 B_0^{(i)} \leq B_1^{(i)} \leq B_0^{(i)} + \frac{(M - \omega_1)\|s_0\|^2}{\operatorname{tr}(E_0^2)} \left(s_0^{(i)}\right)^2. \tag{3.15}$$

Using the similar argument as above, that is, by letting $s_0^{(M)}$ be the largest component (in magnitude) of s_0, then it follows that

$$\min\left(1, \frac{m}{\omega_2}\right)\omega_1 \leq B_1^{(i)} \leq \omega_2 + n(M - \omega_1). \tag{3.16}$$

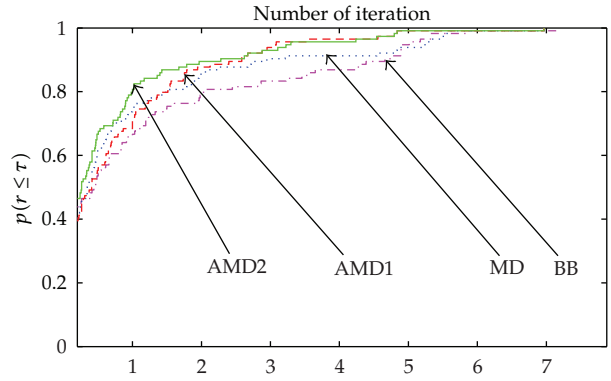

Figure 1: Performance profile based on Iteration for all problems.

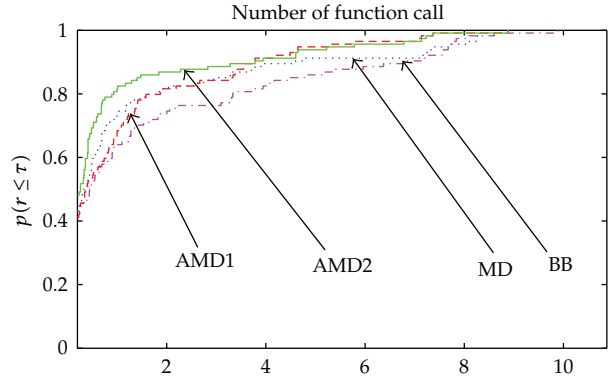

Figure 2: Performance profile based on function call.

Hence, in both cases, $B_1^{(i)}$ is bounded above and below, by some positive constants. Since the upper and lower bound for $B_1^{(i)}$ is, respectively, independent to k, we can proceed by using induction to show that $B_k^{(i)}$ is bounded, for all finite k. □

4. Numerical Results

In this section, we examine the practical performance of our proposed algorithm in comparison with the BB method and standard one-step diagonal gradient-type method (MD). The new algorithms are referred to as AMD1 and AMD2 when $M = I$ and $M = B_k$ are used, respectively. For all methods we employ Armijo line search [11] where $\sigma = 0.9$. All experiments in this paper are implemented on a PC with Core Duo CPU using Matlab 7.0. For each run, the termination condition is that $\|g_k\| \leq 10^{-4}$. All attempts to solve the test problems were limited to a maximum of 1000 iterations. The test problems are chosen from Andrei [13] and Moré et al. [14] collections. The detailed test problem is summarized in Table 1. Our experiments are performed on a set of 36 nonlinear unconstrained problems, and the problems vary in size from $n = 10$ to 10000 variables. Figures 1, 2, and 3 present the Dolan and Moré [15] performance profile for all algorithms subject to the iteration, function call, and CPU time.

Table 1: Test problem and its dimension.

Problem	Dimension	References
Extended Trigonometric, Penalty 1, Penalty 2	$10, \ldots, 10000$	Moré et al. [14]
Quadratic QF2, Diagonal 4, Diagonal 5, Generalized Tridiagonal 1		
Generalized Rosenbrock, Generalized PSC1, Extended Himmelblau		
Extended Three Exponential Terms, Extended Block Diagonal BD1		
Extended PSC1, Raydan 2, Extended Tridiagonal 2, Extended Powell		
Extended Freudenstein and Roth, Extended Rosenbrock	$10, \ldots, 10000$	Andrei [13]
Extended Beale, Broyden Tridiagonal, Quadratic Diagonal Perturbed	$10, \ldots, 1000$	Moré et al. [14]
Perturbed Quadratic, Quadratic QF1, Diagonal 1, Diagonal 2, Hager		
Diagonal 3, Generalized Tridiagonal 2, Almost perturbed Quadratic		
Tridiagonal perturbed quadratic, Full Hessian FH1, Full Hessian FH2		
Raydan 1, EG2, Extended White and Holst	$10, \ldots, 1000$	Andrei [13]

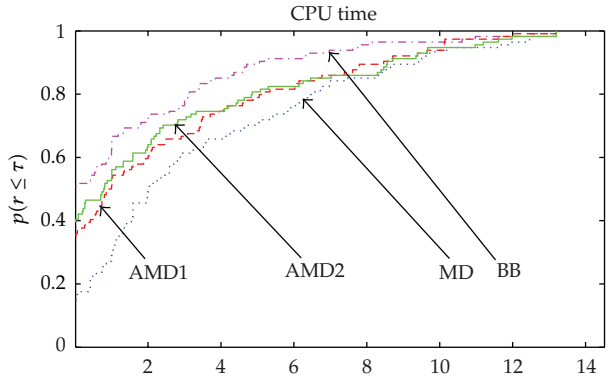

Figure 3: Performance profile based on CPU time per iteration.

From Figure 1, we see that AMD2 method is the top performer, being more successful than other methods in the number of iteration. Figure 2 shows that AMD2 method requires the fewest function calls. From Figure 3, we observe that the AMD2 method is faster than MD and AMD1 methods and needs reasonable time to solve large-scale problems when compared to the BB method. At each iteration, the proposed method does not require more storage than classic diagonal updating methods. Moreover, a higher-order accuracy in approximating the Hessian matrix of the objective function makes AMD method need less iterations and less function evaluation. The numerical results by the tests reported in Figures 1, 2, and 3 demonstrate clearly the new method AMD2 shows significant improvements, when compared with BB, MD, and AMD1. Generally, $M = B_k$ performs better than $M = I$. It is most probably due to the fact that B_k is a better Hessian approximation than the identity matrix I.

5. Conclusion

In this paper, we propose a new two-step diagonal gradient method as view of accumulative approach for unconstrained optimization. The new parameterization for multistep diagonal gradient-type method is developed via employing accumulative approach. The new technique is devised for interpolating curves which are the basis of multistep approach.

Numerical results show that the proposed method is suitable to solve large-scale unconstrained optimization problems and more stable than other similar methods in practical computation. The improvement that our proposed methods bring does come at a complexity cost of $O(n)$ while others are about $O(n^2)$ [9, 10].

References

[1] J. Barzilai and J. M. Borwein, "Two-point step size gradient methods," *IMA Journal of Numerical Analysis*, vol. 8, no. 1, pp. 141–148, 1988.

[2] M. A. Hassan, W. J. Leong, and M. Farid, "A new gradient method via quasi-Cauchy relation which guarantees descent," *Journal of Computational and Applied Mathematics*, vol. 230, no. 1, pp. 300–305, 2009.

[3] W. J. Leong, M. A. Hassan, and M. Farid, "A monotone gradient method via weak secant equation for unconstrained optimization," *Taiwanese Journal of Mathematics*, vol. 14, no. 2, pp. 413–423, 2010.

[4] W. J. Leong, M. Farid, and M. A. Hassan, "Scaling on diagonal Quasi-Newton update for large scale unconstrained Optimization," *Bulletin of the Malaysian Mathematical Sciences Soceity*, vol. 35, no. 2, pp. 247–256, 2012.

[5] M. Y. Waziri, W. J. Leong, M. A. Hassan, and M. Monsi, "A new Newtons method with diagonal jacobian approximation for systems of nonlinear equations," *Journal of Mathematics and Statistics*, vol. 6, pp. 246–252, 2010.

[6] J. E. Dennis, Jr. and H. Wolkowicz, "Sizing and least-change secant methods," *SIAM Journal on Numerical Analysis*, vol. 30, no. 5, pp. 1291–1314, 1993.

[7] M. Farid, W. J. Leong, and M. A. Hassan, "A new two-step gradient-type method for large-scale unconstrained optimization," *Computers & Mathematics with Applications*, vol. 59, no. 10, pp. 3301–3307, 2010.

[8] M. Farid and W. J. Leong, "An improved multi-step gradient-type method for large scale optimization," *Computers & Mathematics with Applications*, vol. 61, no. 11, pp. 3312–3318, 2011.

[9] J. A. Ford and I. A. Moghrabi, "Alternating multi-step quasi-Newton methods for unconstrained optimization," *Journal of Computational and Applied Mathematics*, vol. 82, no. 1-2, pp. 105–116, 1997.

[10] J. A. Ford and S. Tharmlikit, "New implicit updates in multi-step quasi-Newton methods for unconstrained optimisation," *Journal of Computational and Applied Mathematics*, vol. 152, no. 1-2, pp. 133–146, 2003.

[11] L. Armijo, "Minimization of functions having Lipschitz continuous first partial derivatives," *Pacific Journal of Mathematics*, vol. 16, pp. 1–3, 1966.

[12] R. H. Byrd and J. Nocedal, "A tool for the analysis of quasi-Newton methods with application to unconstrained minimization," *SIAM Journal on Numerical Analysis*, vol. 26, no. 3, pp. 727–739, 1989.

[13] N. Andrei, "An unconstrained optimization test functions collection," *Advanced Modeling and Optimization*, vol. 10, no. 1, pp. 147–161, 2008.

[14] J. J. Moré, B. S. Garbow, and K. E. Hillstrom, "Testing unconstrained optimization software," *ACM Transactions on Mathematical Software*, vol. 7, no. 1, pp. 17–41, 1981.

[15] E. D. Dolan and J. J. Moré, "Benchmarking optimization software with performance profiles," *Mathematical Programming A*, vol. 91, no. 2, pp. 201–213, 2002.