Available online at www.sciencedirect.com
ScienceDirect
Mathematics and Computers in Simulation 120 (2016) 24–30
MATHEMATICS AND COMPUTERS IN SIMULATION
www.elsevier.com/locate/matcom

Original articles

Accelerated diagonal gradient-type method for large-scale unconstrained optimization

Mahboubeh Farid

Department of Mathematical and Actuarial Sciences, Lee Kong Chian Faculty of Engineering and Science, Universiti Tunku Abdul Rahman, 43000 Kajang, Selangor, Malaysia

Received 31 July 2012; received in revised form 13 September 2013; accepted 29 December 2014
Available online 16 June 2015

Abstract

In this study, we propose an accelerated diagonal-updating scheme for solving large-scale optimization, where a scaled diagonal matrix is used to approximate the Hessian. We combine an accelerator with the diagonal-updating method to improve the efficiency of the algorithm. This accelerator is employed to ensure that the function value can be reduced significantly at each step. Moreover, the algorithm employs a suitable monotone strategy to guarantee the global convergence of the algorithm. Several numerical results are reported, which demonstrate that the proposed method is promising and more robust than other diagonal updating schemes.

Keywords: Accelerator parameter; Diagonal updating; Large-scale problem; Unconstrained optimization; Weak secant equation

1. Introduction

We consider the unconstrained optimization problem

$$\min f(x), \quad x \in R^n \tag{1}$$

where $f : R^n \longrightarrow R$ is continuously differentiable and its gradient is available. Barzilai and Borwein (BB) [3] proposed a two-point stepsize method for solving this unconstrained optimization problem. The basic idea of the BB method is to choose a scalar α_k such that $\alpha_k I$ is an approximation of the Hessian matrix $\nabla^2 f(x^*)$. To achieve this, a certain quasi-Newton property is imposed and α_k is computed such that $\alpha_k = \frac{s_{k-1}^T s_{k-1}}{s_{k-1}^T y_{k-1}}$, where $s_{k-1} = x_k - x_{k-1}$ and $y_{k-1} = g_k - g_{k-1}$. The BB method has attracted much attention due to its simplicity and low computational requirements, but it suffers from nonmonotonic behavior in each step and it fails to solve ill-conditioned problems [10]. Recently, some variants of diagonal gradient methods were studied in [9,11,13], which use a diagonal matrix as an approximation of the Hessian along the negative gradient direction and a monotone strategy is employed to guarantee a continuous decrease in the function value at each step. However, the monotone strategy employed will only ensure a decrease in the function value at each iteration and the reduction is frequently insignificant. Motivated by

E-mail address: mfarid7@gmail.com.

http://dx.doi.org/10.1016/j.matcom.2014.12.009

this observation, it is desirable to develop an approach that improves the diagonal gradient type methods by decreasing the function value in an effective manner during each step. To achieve this aim, we employ an accelerator parameter (see [2]) in the updating scheme to speed up the convergence to a suitable extent. The remainder of this paper organized as follows. In Section 2, we propose a new gradient scheme, in which we combine the best accelerator with scaled diagonal updating to reduce the function value in each step, and we describe the steps of our new acceleration algorithm. In Section 3, we show that the new algorithm converges globally under some mild assumption. In Section 4, we present computational results obtained using our proposed method, which demonstrate its superior performance compared with other diagonal methods.

2. Accelerated diagonal gradient type method

In this section, we describe the procedure used to construct the new diagonal updating scheme, where we modify the classical gradient method by focusing on reducing the function value. The basis of our approach is to obtain an appropriate level of acceleration, similar to that proposed by Andrei [2], thereby significantly improving the performance of the diagonal updating scheme. The diagonal gradient type method generates a sequence $\{x_k\}$, starting from an initial point $x_0 \in R^n$, using the scheme

$$x_{k+1} = x_k - B_k^{-1} g_k, \tag{2}$$

where x_k is the current iteration, $g_k = \nabla f(x_k)$, and B_k is an approximation of the Hessian in diagonal matrix form, which is selected along the negative direction g_k to minimize $f(x)$. The updated matrix B_{k+1} is restricted to being a diagonal, so it is sufficient for B_{k+1} to satisfy the weak secant equation $s_k^T B_{k+1} s_k = s_k^T y_k$ [6]. Using this assumption, B_{k+1} is updated by the following rule:

$$B_{k+1} = B_k + \frac{s_k^T y_k - s_k^T B_k s_k}{tr(E^2)}, E \tag{3}$$

where $E_k = diag((s_k^{(1)})^2, (s_k^{(2)})^2, \ldots, (s_k^{(n)})^2)$ and tr denotes trace operator. Moreover, a scaling parameter γ_k is considered in the updating scheme to scale B_k at every iteration when B_k have relatively large eigenvalues. Thus,

$$\gamma_k = min\left(1, \frac{s_k^T y_k}{s_k^T B_k s_k}\right) \tag{4}$$

is selected (see [8,12] for details). Therefore, B_{k+1} is redefined as follows:

$$B_{k+1} = \gamma_k B_k + \frac{s_k^T y_k - s_k^T (\gamma_k B_k) s_k}{tr(E^2)}.E. \tag{5}$$

An advantage of the scaling diagonal matrix is that B_{k+1} (5) preserves the positive definiteness property. Moreover, an efficient and simple monotone strategy is employed along the scaled diagonal during updating to guarantee the monotonicity property [12]. In general, the method works well, but the monotone strategy employed is rather conservative for reducing the function values. Thus, the fundamental concept that we introduce is to determine a positive parameter as an accelerator in the updating scheme to overcome this drawback. We propose an accelerated version of the diagonal updating method in order to reduce the significant function value sufficiently during iterations, while the space complexity of our proposed method is maintained as only $O(n)$. Now, we introduce the accelerated diagonal gradient method according to the following scheme:

$$x_{k+1} = x_k - \rho_k B_k^{-1} g_k, \tag{6}$$

where ρ_k is a parameter employed to enhance the behavior of the algorithm. Now, we have

$$f(x_k - B_k^{-1} g_k) = f(x_k) - g_k^T B_k^{-1} g_k + \frac{1}{2}(g_k^T B_k^{-1}) \nabla^2 f(x_k)(B_k^{-1} g_k) + o\left(\|B_k^{-1} g_k\|^2\right), \tag{7}$$