manuscript No.
(will be inserted by the editor)

An efficient gradient method with approximately optimal stepsizes based on regularization models for unconstrained optimization

Zexian Liu[1,*] · Wangli Chu[2] · Hongwei Liu[2]

Received: date / Accepted: date

Abstract It is widely accepted that the stepsize is of great significance to gradient method. An efficient gradient method with approximately optimal stepsizes mainly based on regularization models is proposed for unconstrained optimization. More specifically, if the objective function is not close to a quadratic function on the line segment between the current and latest iterates, regularization model is exploited carefully to generate approximately optimal stepsize. Otherwise, quadratic approximation model is used. In addition, when the curvature is non-positive, special regularization model is developed. The convergence of the proposed method is established under some weak conditions. Extensive numerical experiments indicated the proposed method is very promising. Due to the surprising efficiency, we believe that gradient methods with approximately optimal stepsizes can become strong candidates for large-scale unconstrained optimization.

Keywords Approximately optimal stepsize. Gradient method. Regularization method. Barzilai-Borwein (BB) method. Global convergence.

Mathematics Subject Classification (2000) 90C06 · 65K

1 Introduction

We consider the following unconstrained optimization problem:

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

where $f : R^n \to R$ is continuously differentiable and its gradient is denoted by g. The gradient method for solving (1) has the following form

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

Zexian Liu(✉), e-mail: liuzexian2008@163.com
School of Mathematics and Statistics, Guizhou University, Guiyang, 550025, China

Wangli Chu, e-mail: chuwangli123@163.com; Hongwei Liu, e-mail:hwliuxidian@163.com
School of Mathematics and Statistics, Xidian University, Xi'an, 710126, China

This provisional PDF is the accepted version. The article should be cited as: RAIRO: RO, doi: 10.1051/ro/2022107

where α_k is the stepsize and $g_k = \nabla f(x_k)$. Throughout this paper, $f_k = f(x_k)$, $s_{k-1} = x_k - x_{k-1}$, $y_{k-1} = g_k - g_{k-1}$ and $\|.\|$ denotes the Euclidean norm.

It is widely accepted that the stepsize is of great significance to the theory and numerical performance of gradient method, and the stepsize is the core problem of gradient method. The classical steepest descent method [1], where the stepsize is given by $\alpha_k^{SD} = \arg\min_{\alpha>0} f(x_k - \alpha g_k)$, is badly affected by ill conditioning and thus converges slowly [2]. In 1988, Barzilai and Borwein [3] proposed a two-point gradient method (BB method), where the famous stepsize (BB stepsize) is given by

$$\alpha_k^{BB_1} = \frac{\|s_{k-1}\|^2}{s_{k-1}^T y_{k-1}} \quad \text{or} \quad \alpha_k^{BB_2} = \frac{s_{k-1}^T y_{k-1}}{\|y_{k-1}\|^2}. \tag{3}$$

Due to the simplicity and nice numerical efficiency, the BB method has received extensive attention. The BB method has been shown to be globally [4] and R-linearly [5] convergent for any dimensional strictly convex quadratic functions. In 2021, Li and Sun [6] presented an interesting improved R-linear convergence result of the BB method. Raydan [8] proposed the global BB method by incorporating the nonmonotone line search (GLL line search) [9]. Dai et al. [7] presented a quite efficient gradient method by adaptively choosing the BB stepsizes. Dai et al. [10] viewed the BB stepsize from a new angle and constructed a quadratic model and a conic model to derive two stepsizes for gradient methods. In 2018, Liu et al. [11] viewed the stepsize $\alpha_k^{BB_1}$ from the approximation model and introduced a new type of stepsize called approximately optimal stepsize for gradient method.

Definition 1.1 [11] Suppose f is continuously differentiable, and let $\phi_k(\alpha)$ be an approximation model of $f(x_k - \alpha g_k)$. A positive constant α_k^{AOS} is called **approximately optimal stepsize** associated to $\phi_k(\alpha)$ for gradient method if α_k^{AOS} satisfies

$$\alpha_k^{AOS} = \arg\min_{\alpha>0} \phi_k(\alpha). \tag{4}$$

From (4), we can easily obtain the following simple facts:

(i)If $\phi_k(\alpha) = f(x_k - \alpha g_k)$, then the resulted approximately optimal stepsize corresponds to Cauchy stepsize. This is the reason that we call the stepsize (4) approximately optimal stepsize.

(ii)If $\phi_k(\alpha) = f_k - \alpha\|g_k\|^2 + \frac{1}{2}\alpha^2 g_k^T \left(\frac{s_{k-1}^T y_{k-1}}{\|s_{k-1}\|^2} I\right) g_k$, then the resulted approximately optimal stepsize corresponds to the BB stepsize $\alpha_k^{BB_1}$.

(iii)For any stepsize $\alpha_k > 0$, let $\phi_k(\alpha) = f_k - \alpha\|g_k\|^2 + \frac{1}{2}\alpha^2 g_k^T \left(\frac{1}{\alpha_k} I\right) g_k$, it is easy to see that the resulted approximately optimal stepsize is exactly α_k. As a result, all existing stepsizes for gradient methods can be treated as approximately optimal stepsizes in this sense.

Some gradient methods with approximately optimal stepsizes [12, 13] were proposed, and the numerical experiments in [12, 13] indicated that these gradient methods are very efficient. Gradient methods with approximately optimal stepsizes have illustrated powerful potentiality for unconstrained optimization.

In addition, based on a fourth order conic model and some modified secant equations, Biglari and Solimanpur [14] presented some modified BB methods. Recently, motivated by Yuan's stepsize [15], Huang et al. [16] equipped the Barzilai and Borwein method with two dimensional quadratic termination property and proposed a novel stepsize for gradient method (HDL, corresponding to Algorithm 3.1 in [16]) for general unconstrained optimization. More modified BB methods can be found in [17–20].

Contributions. According to Definition 1.1, it is not difficult to see that the effectiveness of approximately optimal stepsize relies heavily on the approximation model $\phi_k(\alpha)$. To obtain more efficient gradient methods with approximately optimal stepsizes, one should take full advantage of the properties of f at x_k to exploit suitable approximation models including quadratic models and non-quadratic models for deriving approximately optimal stepsize. In the paper, we present an efficient gradient method with approximately optimal stepsizes based on regularization models for unconstrained optimization. In the proposed method, if the objective function f is not close to a quadratic function on the line segment between x_{k-1} and x_k, then a regularization model is exploited to generate approximately optimal stepsizes. Otherwise, a quadratic approximation model is used to derive approximately optimal stepsize. In addition, when $s_{k-1}^T y_{k-1} \leq 0$, a special regularization model is developed carefully. The global convergence of the proposed method is analyzed. The numerical results indicate that the proposed method is superior to the HDL method [16] and other efficient gradient methods, and is competitive to two famous conjugate gradient software packages CGOPT (1.0) [21] and CG_DESCENT (5.0) [22] for the 145 test problems in the CUTEr library [23], and has significant improvement over CGOPT (1.0) [21] and CG_DESCENT (5.0) [22] for the 80 test problems mainly from [24].

The rest of the paper is organized as follows. In Section 2, some approximation models including regularization models and quadratic models are exploited to generate approximately optimal stepsizes for gradient method. In Section 3, an efficient gradient method with the approximately optimal stepsizes is described in detail. The global convergence of the proposed method is analyzed in Section 4. In Section 5, some numerical results are presented. Conclusion and discussion are given in the last section.

2 Derivation of Approximately Optimal Stepsizes

Based on the properties of f at the current iterate x_k, some approximation models including regularization models and quadratic models are exploited carefully to derive approximately optimal stepsizes for gradient method in the section.

As mentioned above, the effectiveness of approximately optimal stepsize relies heavily on approximation model $\phi_k(\alpha)$. So we design carefully suitable approximation models mainly based on the properties of f at x_k. The choices of approximation models are from the following observation.

Define

$$\mu_k = \left| \frac{2(f_{k-1} - f_k + g_k^T s_{k-1})}{s_{k-1}^T y_{k-1}} - 1 \right|. \tag{5}$$

According to [11], μ_k is an important criterion for measuring the degree of f to approximate quadratic function. If the condition [10, 12]

$$\mu_k \leq c_1 \quad \text{or} \quad \max\{\mu_k, \mu_{k-1}\} \leq c_2 \tag{6}$$

holds, then f might be close to a quadratic function on the line segment between x_{k-1} and x_k. Here $0 < c_1 < c_2$,

When f is close to a quadratic function on the line segment between x_{k-1} and x_k, quadratic approximation model is certainly preferable. However, if the objective function f possesses high non-linearity, then quadratic models might not work very well [25, 26], so some nonquadratic approximation models should be considered in this case. In recent years, regularization algorithms, which are defined as the standard quadratic model plus a regularization term, have been proposed for unconstrained optimization [27]. An adaptive regularization algorithm using cubics (ARC) was proposed by Cartis et al. [27]. The trial step in ARC algorithm [27] is computed by minimizing the following regularization model:

$$m_k(d) = f(x_k) + g_k^T d + \frac{1}{2} d^T B_k d + \frac{1}{3}\sigma_k \|d\|^3, \tag{7}$$

where B_k is a symmetric approximation to the Hessian matrix and $\sigma_k > 0$ is a regularization parameter. And the numerical results in [28] indicated that ARC algorithm is quite efficient. More advance about regularization algorithms can be referred to [29–31]. Regularization algorithms have become an alternative to trust region and line search schemes [27]. All of this indicates that when f is not close to a quadratic function around x_k, regularization models might serve better than quadratic models generally.

Motivated by the above observation, we consider the approximation model (7), and derive approximately optimal stepsizes for gradient methods in the following four cases based on the sign of $s_{k-1}^T y_{k-1}$ and the condition (6).

Case I. $s_{k-1}^T y_{k-1} > 0$ holds and the condition (6) does not hold.

In the case, the objective function f might be not close to a quadratic function on the line segment between x_{k-1} and x_k, we thus use the regularization model (7) with $d = -\alpha g_k$:

$$\phi_1(\alpha) = f(x_k) - \alpha g_k^T g_k + \frac{1}{2}\alpha^2 g_k^T B_k g_k + \frac{1}{3}\alpha^3 \sigma_k \|g_k\|^3. \tag{8}$$

Taking account of the computational cost and storage, B_k is generated by imposing the modified Broyden-Fletcher-Goldfarb-Shanno (BFGS) update formula [32] on a scalar matrix D_k:

$$B_k = D_k - \frac{D_k s_{k-1} s_{k-1}^T D_k}{s_{k-1}^T D_k s_{k-1}} + \frac{\overline{y}_{k-1}\overline{y}_{k-1}^T}{s_{k-1}^T \overline{y}_{k-1}}, \tag{9}$$

where $\bar{y}_{k-1} = y_{k-1} + \frac{\bar{r}_k}{\|s_{k-1}\|^2} s_{k-1}$ and $\bar{r}_k = 3(g_k + g_{k-1})^T s_{k-1} + 6(f_{k-1} - f_k)$. Here we take D_k as $D_k = \xi_0 \frac{y_{k-1}^T y_{k-1}}{s_{k-1}^T y_{k-1}} I$, where $\xi_0 \geq 1$. If f is twice continuously differentiable, then there exists $\mu_1 \in [0, 1]$ such that

$$\bar{r}_k = 3(s_{k-1}^T y_{k-1} - s_{k-1}^T \nabla^2 f(x_{k-1} + \mu_1 s_{k-1}) s_{k-1}). \tag{10}$$

Therefore, to improve the numerical performance we restrict $\bar{r}_k$ as

$$\bar{r}_k = \min\left\{\max\left\{\bar{r}_k, -\xi_1 s_{k-1}^T y_{k-1}\right\}, \xi_1 s_{k-1}^T y_{k-1}\right\}, \tag{11}$$

where $0 < \xi_1 < 0.1$.

As for the choice of regularization parameter σ_k in (8), we determine it as follow. The regularization parameter is significant to the effectiveness of regularization model. However, it is universally acknowledged that it is challenging to determine a proper regularization parameter σ_k. Some ways including the interpolation condition and the trust-region strategy [33, 34] were developed to determine the regularization parameter σ_k. Here we use the interpolation condition to determine the regularization parameter:

$$f_{k-1} = f_k - g_k^T s_{k-1} + \frac{1}{2} s_{k-1}^T B_k s_{k-1} + \frac{\sigma_k}{3} \|s_{k-1}\|^3,$$

which implies that

$$\sigma_k = \frac{3\left(f_{k-1} - f_k + g_k^T s_{k-1} - \frac{1}{2} s_{k-1}^T y_{k-1}\right)}{\|s_{k-1}\|^3}. \tag{12}$$

To improve the numerical performance and make it to be positive, we take the following truncated form of (12):

$$\sigma_k = \max\left\{\min\left\{|\sigma_k|, \sigma_{\max}\right\}, \sigma_{\min}\right\}, \tag{13}$$

where $0 < \sigma_{\min} < \sigma_{\max}$.

It is not difficult to obtain the following lemma.

Lemma 2.1. *Suppose that* $s_{k-1}^T y_{k-1} > 0$. *Then,* $s_{k-1}^T \bar{y}_{k-1} > 0$ *and* B_k *is symmetric and positive definite.*

By imposing $\frac{d\phi_1}{d\alpha} = 0$, we obtain the equation $-g_k^T g_k + \alpha g_k^T B_k g_k + \alpha^2 \sigma_k \|g_k\|^3 = 0$. Since

$$\varDelta_1 = (g_k^T B_k g_k)^2 + 4\sigma_k \|g_k\|^5 > 0, \tag{14}$$

the above equation has a positive root and a negative root. According to Definition 1.1, it is not difficult to verify that the positive root is the approximately optimal stepsize, namely,

$$\bar{\alpha}_k^{AOS(1)} = \frac{2\|g_k\|^2}{\sqrt{\varDelta_1} + g_k^T B_k g_k}. \tag{15}$$

where B_k is given by (9) with (11).

It is observed by numerical experiments that the bound $\left[\alpha_k^{BB_2}, \alpha_k^{BB_1}\right]$ for $\bar{\alpha}_k^{AOS(1)}$ is very preferable. Therefore, if $s_{k-1}^T y_{k-1} > 0$ holds and the condition (6) does not hold, then we take the following truncated approximately optimal stepsize

$$\alpha_k^{AOS(1)} = \max\left\{\min\left\{\bar{\alpha}_k^{AOS(1)}, \alpha_k^{BB_1}\right\} \alpha_k^{BB_2}\right\} \tag{16}$$

for gradient method.

Case II. $s_{k-1}^T y_{k-1} > 0$ and the condition (6) hold.

In the case, the objective function f might be close to a quadratic function on the line segment between x_{k-1} and x_k, we thus consider the following quadratic approximation model:

$$\phi_2(\alpha) = f(x_k) - \alpha g_k^T g_k + \frac{1}{2}\alpha^2 g_k^T B_k g_k, \tag{17}$$

where B_k is given by (9) with (11) for simplicity. It follows from Lemma 2.1 that B_k is symmetric and positive definite. By imposing $\frac{d\phi_2}{d\alpha} = 0$, we can easily obtain the approximately optimal stepsize

$$\bar{\alpha}_k^{AOS(2)} = \frac{g_k^T g_k}{g_k^T B_k g_k} = \frac{\|g_k\|^2}{\frac{\xi_1 \|y_{k-1}\|^2}{s_{k-1}^T y_{k-1}}\left(\|g_k\|^2 - \frac{\left(g_k^T s_{k-1}\right)^2}{\|s_{k-1}\|^2}\right) + \frac{\left(g_k^T \bar{y}_{k-1}\right)^2}{s_{k-1}^T \bar{y}_{k-1}}}. \tag{18}$$

Similar to **Case I**, if $s_{k-1}^T y_{k-1} > 0$ and the condition (6) hold, then we take the truncated approximately optimal stepsize

$$\alpha_k^{AOS(2)} = \max\left\{\min\left\{\bar{\alpha}_k^{AOS(2)}, \alpha_k^{BB_1}\right\}, \alpha_k^{BB_2}\right\} \tag{19}$$

for gradient method.

Case III. $s_{k-1}^T y_{k-1} \le 0$ and the condition (20) hold.

When $s_{k-1}^T y_{k-1} \le 0$, the BB stepsizes or the approximately optimal stepsizes described above can not be used, and thus it is difficult to determine suitable stepsize for gradient method. In some modified BB methods [10, 14], the stepsize is usually set simply to $\alpha_k = 10^{30}$ when $s_{k-1}^T y_{k-1} \le 0$. As a result, it will cause large computational cost for seeking a suitable stepsize in a line search for gradient method.

It follows from $s_{k-1}^T y_{k-1} \le 0$ that $0 < \frac{\|g_{k-1}\|}{\|g_k\|} \le 1$. Consequently, if the following condition

$$\xi_2 \le \frac{\|g_{k-1}\|^2}{\|g_k\|^2} \le 1 \tag{20}$$

holds, where $0 < \xi_2 < 1$ is close to 1, then g_k and g_{k-1} tend to be collinear and are approximately equal. In the case, we can use g_{k-1} to approximate g_k, which will be useful for constructing approximation model, as described below.

Suppose for the moment that f is twice continuously differentiable, we consider the following regularization model:

$$\phi(\alpha) = f_k - \alpha g_k^T g_k + \frac{1}{2}\alpha^2 g_k^T \nabla^2 f(x_k) g_k + \frac{\sigma_k}{3}\alpha^3 \|g_k\|^3. \tag{21}$$

When the condition (20) holds, we use $g_{k-1}^T \nabla^2 f(x_k) g_{k-1}$ to approximate $g_k^T \nabla^2 f(x_k) g_k$ and thus get that

$$g_k^T \nabla^2 f(x_k) g_k \approx g_{k-1}^T \nabla^2 f(x_k) g_{k-1} \approx \frac{\left|(g(x_k + \alpha_{k-1} g_{k-1}) - g(x_k))^T g_{k-1}\right|}{\alpha_{k-1}} = \frac{\left|s_{k-1}^T y_{k-1}\right|}{\alpha_{k-1}^2}, \tag{22}$$

which yields the following approximation model:

$$\phi_3(\alpha) = f_k - \alpha g_k^T g_k + \frac{1}{2}\alpha^2 \frac{\left|s_{k-1}^T y_{k-1}\right|}{\alpha_{k-1}^2} + \frac{\sigma_k}{3}\alpha^3 \|g_k\|^3.$$

As for the choice of regularization parameter in the regularization model, similarly to **Case I**, we also use the interpolation condition to determine the regularization parameter σ_k:

$$f_{k-1} = f_k - g_k^T s_{k-1} + \frac{1}{2} s_{k-1}^T y_{k-1} + \frac{\sigma_k}{3} \|s_{k-1}\|^3,$$

which implies that

$$\sigma_k = \frac{3\left(f_{k-1} - f_k + g_k^T s_{k-1} - \frac{1}{2} s_{k-1}^T y_{k-1}\right)}{\|s_{k-1}\|^3}.$$

To improve the numerical performance and make it to be positive, we take the following truncation form:

$$\sigma_k = \max\left\{\min\left\{|\sigma_k|, \sigma_{\max}\right\}, \sigma_{\min}\right\}, \tag{23}$$

where $0 < \sigma_{\min} < \sigma_{\max}$ are the same as that in (13).

By imposing $\frac{d\phi_3}{d\alpha} = 0$, we get the equation $-\|g_k\|^2 + \alpha \frac{\left|s_{k-1}^T y_{k-1}\right|}{\alpha_{k-1}^2} + \alpha^2 \sigma_k \|g_k\|^3 = 0$. Since

$$\Delta_2 = \frac{\left|s_{k-1}^T y_{k-1}\right|^2}{\alpha_{k-1}^4} + 4\sigma_k \|g_k\|^5 > 0,$$

the above equation has a positive root and a negative root. By Definition 1.1, it is not difficult to verify that the positive root is the approximately optimal stepsize, namely,

$$\alpha_k^{AOS(3)} = \frac{2\|g_k\|^2 \alpha_{k-1}^2}{\sqrt{\left|s_{k-1}^T y_{k-1}\right|^2 + 4\alpha_{k-1}^4 \sigma_k \|g_k\|^5} + \left|s_{k-1}^T y_{k-1}\right|}. \tag{24}$$

Case IV. $s_{k-1}^T y_{k-1} \le 0$ holds and the condition (20) does not hold

It also has been shown that if $\alpha_k^{BB_1}$ is reused in a cyclic fashion, then the convergence rate is accelerated [35]. It appears that α_{k-1} may be helpful for determining the current stepsize α_k. Therefore, we take $\xi_3 \alpha_{k-1}$ as the stepsize, where $\xi_3 > 0$. In actual, the stepsize can also be regarded as an approximately optimal stepsize. Substituting $B_k = \frac{1}{\xi_3 \alpha_{k-1}} I$ into (17) yields the following approximation model

$$\phi_4(\alpha) = f(x_k) - \alpha g_k^T g_k + \frac{1}{2}\alpha^2 g_k^T \left(\frac{1}{\xi_3 \alpha_{k-1}} I \right) g_k. \tag{25}$$

By imposing $\frac{d\phi_4}{d\alpha} = 0$, we obtain the approximately optimal stepsize:

$$\alpha_k^{AOS(4)} = \xi_3 \alpha_{k-1}. \tag{26}$$

3 Gradient Method with Approximately Optimal Stepsizes

We describe the gradient method with approximately optimal stepsizes in the section.

The famous nonmonotone line search (GLL line search) [9] was firstly incorporated into the BB method [8]. Though GLL line search works well in many cases, there are some drawbacks. For example, the numerical performance depends heavily on the choice of a pre-fixed memory constant M. To overcome the above drawbacks, another nonmonotone Armijo line search (Zhang-Hager line search) was proposed by Zhang and Hager [36], which is defined as

$$f(x_k - \alpha g_k) \le C_k - \delta\alpha \|g_k\|^2, \tag{27}$$

where $0 < \delta < 1$,

$$Q_0 = 1,\ C_0 = f(x_0),\ Q_{k+1} = \eta_k Q_k + 1,\ C_{k+1} = (\eta_k Q_k C_k + f(x_{k+1}))/Q_{k+1},\ \ 0 < \eta_k \le 1. \tag{28}$$

It is observed that Zhang-Hager line search [36] is usually preferable for modified BB methods. To improve the numerical performance and obtain nice convergence, we take η_k as :

$$\eta_k = \begin{cases} c, & \mathrm{mod}(k,n) = n-1, \\ 1, & \mathrm{mod}(k,n) \neq n-1, \end{cases} \tag{29}$$

where $0 < c < 1$ and $\mathrm{mod}(k,n)$ represents the residue for k modulo n. As a result, Zhang-Hager line search with (29) and the following strategy [37]:

$$\alpha = \begin{cases} \overline{\alpha}, & \text{if } \alpha > 0.1\alpha_k^{(0)} \text{ and } \overline{\alpha} \in [0.1\alpha_k^{(0)}, 0.9\alpha], \\ 0.5\alpha, & \text{otherwise} \end{cases} \tag{30}$$

is used in the our method. Here $\alpha_k^{(0)}$ is approximately optimal stepsize described in Section 2 and $\overline{\alpha}$ is obtained by a quadratic interpolation at x_k and $x_k - \alpha g_k$,

We describe the gradient method with approximately optimal stepsizes in detail.

Algorithm 1 Gradient Method with Approximately Optimal Stepsizes (GM_AOS (Reg p=3))

Step 0. Initialization. Given $x_0 \in R^n$, $\varepsilon > 0$, δ, c, c_1, c_2 $\alpha_{\max}$, $\alpha_{\min}$, α_0^0, $\sigma_{\min}$, $\sigma_{\max}$, ξ_0, ξ_1, ξ_2, ξ_3. Set $Q_0 = 1$, $C_0 = f_0$ and $k = 0$.
Step 1. If $\|g_k\|_\infty \le \varepsilon$, then stop.
Step 2. Compute approximately optimal stepsize.
  2.1 If $k = 0$, then set $\alpha = \alpha_0^{(0)}$ and go to Step 3.
  2.2 If $s_{k-1}^T y_{k-1} > 0$ holds and the condition (6) does not hold, then compute α_k by (16).
  2.3 If $s_{k-1}^T y_{k-1} > 0$ holds and the condition (6) holds, then compute α_k by (19).
  2.4 If $s_{k-1}^T y_{k-1} \le 0$ holds and the condition (20) holds, then compute α_k by (24).
  2.5 If $s_{k-1}^T y_{k-1} \le 0$ holds and the condition (20) does not hold, then compute α_k by (26).
  2.6 Set $\alpha_k^{(0)} = \max\{\min\{\alpha_k, \alpha_{\max}\}, \alpha_{\min}\}$ and $\alpha = \alpha_k^{(0)}$.
Step 3. Line search. If (27) holds, then go to Step 4, otherwise update α by (30) and go to Step 3.
Step 4. Update Q_{k+1}, C_{k+1} and η_k by (28) and (29).
Step 5. Set $\alpha_k = \alpha$, $x_{k+1} = x_k - \alpha_k g_k$, $k = k + 1$, and go to Step 1.

4 Convergence Analysis

In the section the global convergence of GM_AOS (Reg p=3) is analyzed under some weak assumptions: (D1) f is continuously differentiable on R^n; (D2) f is bounded below on R^n; (D3) The gradient g is **uniformly continuous** on R^n.

We first give two lemma, which is important to the convergence.

Lemma 4.1 *For Q_k in (28) , we have $Q_{k+1} \le 1 + \dfrac{n}{1-c}$.*

Proof It follows from (28) that

$$Q_{k+1} = 1 + \sum_{j=0}^{k} \prod_{i=0}^{j} \eta_{k-i},$$

which together with (29) suggests that

$$Q_{k+1} = \begin{cases} 1 + n \sum\limits_{i=1}^{(k+1)/n} c^i\,, & \text{if } \bmod(k,n) = n-1, \\ 1 + \left(1 + \bmod(k,n) + n \sum\limits_{i=1}^{\lfloor k/n \rfloor} c^i\right)\,, & \text{if } \bmod(k,n) \ne n-1, \end{cases} \tag{31}$$

where $\lfloor \cdot \rfloor$ is the floor function.

By (31) and $0 < c < 1$, we obtain that

$$Q_{k+1} \leqslant 1 + \left(n + n \sum_{i=1}^{\lfloor k/n \rfloor + 1} c^i\right) \leqslant 1 + \left(n + n \sum_{i=1}^{k+1} c^i\right) = 1 + n \sum_{i=0}^{k+1} c^i \leqslant 1 + \frac{n}{1-c},$$

which completes the proof. □

Lemma 4.2 *Suppose that the assumptions (D1), (D2) and (D3) hold. Then,*

$$f_{k+1} \le C_{k+1} \le C_k. \tag{32}$$

Proof According to (27) and (28), we have

$$C_{k+1} = \frac{\eta_k Q_k C_k + f_{k+1}}{Q_{k+1}} = C_k + \frac{f_{k+1} - C_k}{Q_{k+1}} \le C_k$$

and

$$C_{k+1} = \frac{\eta_k Q_k C_k + f_{k+1}}{Q_{k+1}} = \frac{\eta_k Q_k}{\eta_k Q_k + 1} C_k + \frac{1}{\eta_k Q_k + 1} f_{k+1} \ge \frac{\eta_k Q_k}{\eta_k Q_k + 1} f_{k+1} + \frac{1}{\eta_k Q_k + 1} f_{k+1} = f_{k+1}.$$

As a result, the inequality (32) holds. The proof is completed. □

The above lemma implies that the sequence $\{C_k\}$ is convergent.

Theorem 4.1 *Suppose that the assumptions (D1), (D2) and (D3) hold, and let $\{x_k\}$ be the sequence generated by GM_AOS (Reg p=3). Then,*

$$\lim_{k \to \infty} \|g_k\| = 0. \tag{33}$$

Proof By (27) and (28), we obtain that

$$C_{k+1} = C_k + \frac{f_{k+1} - C_k}{Q_{k+1}} \le C_k - \frac{\delta \alpha_k \|g_k\|^2}{Q_{k+1}},$$

which together with Lemma 4.1 implies that

$$\frac{\delta}{1 + n/(1-c)} \alpha_k \|g_k\|^2 \le \frac{\delta \alpha_k \|g_k\|^2}{Q_{k+1}} \le C_k - C_{k+1}. \tag{34}$$

It then follows from Lemma 4.2 and assumptions (D2) that

$$\lim_{k \to \infty} \alpha_k \|g_k\|^2 = 0. \tag{35}$$

We suppose, by way of contradiction, that there exists a subsequence $\{x_{k_j}\}$ such that

$$\lim_{j \to \infty} \|g_{k_j}\| = l > 0. \tag{36}$$

Denote

$$\bar{\varepsilon} = \begin{cases} l/2, & \text{if } l < +\infty, \\ 1/2, & \text{otherwise.} \end{cases}$$

It follows from (36) that there exists a positive integer j_0 such that

$$\|g_{k_j}\| > \bar{\varepsilon}, \quad \forall j > j_0. \tag{37}$$

Therefore, we obtain from (35) that $\lim_{j \to \infty} \alpha_{k_j} = 0$ and

$$\lim_{j \to \infty} \alpha_{k_j}^2 \|g_{k_j}\|^2 = 0. \tag{38}$$

By (30), we know that there exists $\bar{\delta}_{k_j} \in [0.1, 0.9]$ such that

$$f\left(x_{k_j} - \frac{\alpha_{k_j}}{\bar{\delta}_{k_j}} g_{k_j}\right) > C_{k_j} - \delta \frac{\alpha_{k_j}}{\bar{\delta}_{k_j}} \left\|g_{k_j}\right\|^2. \tag{39}$$

Combining (39) and $f\left(x_{k_j} - \alpha_{k_j} g_{k_j}\right) \leq C_{k_j} - \delta \alpha_{k_j} \left\|g_{k_j}\right\|^2$ yields

$$f\left(x_{k_j} - \frac{\alpha_{k_j}}{\bar{\delta}_{k_j}} g_{k_j}\right) - f\left(x_{k_j} - \alpha_{k_j} g_{k_j}\right) > -\delta\left(\frac{1}{\bar{\delta}_{k_j}} - 1\right) \alpha_{k_j} \left\|g_{k_j}\right\|^2.$$

It follows from the mean-value theorem that there exists $w_{k_j} \in [0,1]$ such that

$$f\left(x_{k_j} - \frac{\alpha_{k_j}}{\bar{\delta}_{k_j}} g_{k_j}\right) - f\left(x_{k_j} - \alpha_{k_j} g_{k_j}\right) = -\left(\frac{1}{\bar{\delta}_{k_j}} - 1\right) \alpha_{k_j} g\left(u_{k_j}\right)^T g_{k_j},$$

where $u_{k_j} = x_{k_j} - \left[1 + w_{k_j}\left(1/\bar{\delta}_{k_j} - 1\right)\right] \alpha_{k_j} g_{k_j}$. Thus, we get that

$$-\left(\frac{1}{\bar{\delta}_{k_j}} - 1\right) \alpha_{k_j} g\left(u_{k_j}\right)^T g_{k_j} > -\delta\left(\frac{1}{\bar{\delta}_{k_j}} - 1\right) \alpha_{k_j} \left\|g_{k_j}\right\|^2,$$

which implies that $\left(g_{k_j} - g\left(u_{k_j}\right)\right)^T \frac{g_{k_j}}{\left\|g_{k_j}\right\|} > (1-\delta)\left\|g_{k_j}\right\|$. According to (37), we know that

$$\left\|g_{k_j} - g(u_{k_j})\right\| \geq \left(g_{k_j} - g(u_{k_j})\right)^T \frac{g_{k_j}}{\left\|g_{k_j}\right\|} > (1-\delta)\left\|g_{k_j}\right\| > (1-\delta)\bar{\varepsilon}, \quad \forall j > j_0. \tag{40}$$

It follows from (35), (38) and $1 \leq 1 + w_{k_j}\left(1/\bar{\delta}_{k_j} - 1\right) \leq 10$ that

$$\lim_{j \to +\infty} \left[w_{k_j}\left(1/\bar{\delta}_{k_j} - 1\right) + 1\right] \alpha_{k_j} \left\|g_{k_j}\right\| \to 0. \tag{41}$$

Since the gradient g is uniformly continuous, for $\frac{(1-\delta)\bar{\varepsilon}}{2}$, one can find $\zeta > 0$ depending only on $\frac{(1-\delta)\bar{\varepsilon}}{2}$ such that $\left\|g_{k_j} - g\left(u_{k_j}\right)\right\| \leq \frac{(1-\delta)}{2}\bar{\varepsilon}$ holds whenever $\left\|x_{k_j} - u_{k_j}\right\| = \left[w_{k_j}\left(1/\bar{\delta}_{k_j} - 1\right) + 1\right] \alpha_{k_j} \left\|g_{k_j}\right\| < \zeta$. By (41), we know that there exists an integer $j_1 > 0$ such that

$$\left\|x_{k_j} - u_{k_j}\right\| = \left[w_{k_j}\left(1/\bar{\delta}_{k_j} - 1\right) + 1\right] \alpha_{k_j} \left\|g_{k_j}\right\| < \zeta$$

holds for any $j > j_1$. As a result, $\left\|g_{k_j} - g\left(u_{k_j}\right)\right\| \leq \frac{(1-\delta)}{2}\bar{\varepsilon}$ holds for any $j > j_1$, which contradicts (40) when $j \geq \max\{j_0, j_1\}$. Therefore, there no exists a subsequence $\left\{x_{k_j}\right\}$ satisfying (36), which implies (33). The proof is completed. □

5 Numerical Experiments

We compare GM_AOS (Reg p=3) with GM_AOS (1.2) [13], the BB method, CGOPT (1.0) [21], CG_DESCENT (5.0) [22] and HDL method [16] (corresponding to Algorithm 3.1 in [16]) in the section. It is widely accepted that CGOPT [21] and CG_DESCENT [22] are the two most famous conjugate gradient software packages. The BB method, GM_AOS (1.2) [13] and GM_AOS (Reg p=3) were implemented by C code, and the C codes of CG_DESCENT (5.0) and CGOPT (1.0) can be downloaded from Hager's homepage: http://users.clas.ufl.edu/hager/papers/Software and Dai's homepage: http://lsec.cc.ac.cn/~dyh/software.html, respectively. The Matlab code of HDL can be also found in Dai's homepage. Two test sets were used, which include the 145 test problems in the CUTEr library [23] (we call it CUTEr145 for short) and the 80 test problems mainly from [24] collected by Andrei (we call it Andr80 for short), respectively. The two test sets can be found in Hager's homepage: http://users.clas.ufl.edu/hager/papers/CG/results6.0.txt and Andrei's homepage: http://camo.ici.ro/neculai/AHYBRIDM, respectively. The dimensions of the test problem in the test set CUTEr145 are default and the dimension of each test problem in the test set Andr80 is set to 10,000. All numerical experiments were done in Ubuntu 10.04 LTS in a VMware Workstation 10.0 installed in Win 10.

We choose the following parameters for GM_AOS (Reg p=3): $\varepsilon = 10^{-6}$, $\alpha_{\min} = 10^{-30}$, $\alpha_{\max} = 10^{30}$, $\xi_0 = 1.07$, $\xi_1 = 5 \times 10^{-5}/3$, $\xi_2 = 0.8$, $\xi_3 = 5$, $\sigma_{\min} = 10^{-30}$, $\sigma_{\max} = 10^3$, $\delta = 10^{-4}$, $c_1 = 10^{-9}$, $c_2 = 10^{-7}$, $c = 0.99$ and

$$\alpha_0 = \begin{cases} 2\frac{|f_0|}{\|g_0\|^2}, & \text{if} \|x_0\|_\infty < 10^{-30} \text{ and } |f_0| \geq 10^{-30}, \\ 1.0, & \text{if} \|x_0\|_\infty < 10^{-30} \text{ and } |f_0| < 10^{-30}, \\ \min\left\{1.0, \max\left\{\frac{\|x_0\|_\infty}{\|g_0\|_\infty}, \frac{1}{\|g_0\|_\infty}\right\}\right\}, & \text{if} \|x_0\|_\infty \geq 10^{-30} \text{ and } \|g_0\|_\infty \geq 10^7, \\ \min\left\{1.0, \frac{\|x_0\|_\infty}{\|g_0\|_\infty}\right\}, & \text{if} \|x_0\|_\infty \geq 10^{-30} \text{ and } \|g_0\|_\infty < 10^7. \end{cases}$$

GM_AOS (1.2) [13] and the BB method used the same line search as that in GM_AOS (Reg p=3). CGOPT (1.0), CG_DESCENT (5.0) and HDL used all default settings of parameters but the stopping conditions. Each test method is terminated if $\|g_k\|_\infty \leq 10^{-6}$ or the iterations exceeds 140,000.

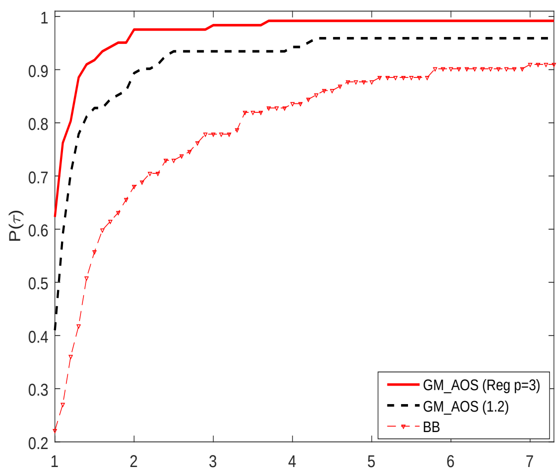


Fig. 1 N_{iter}(CUTEr145)

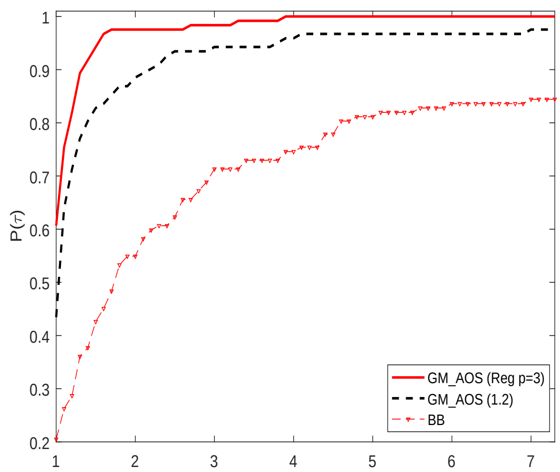


Fig. 2 N_f(CUTEr145)

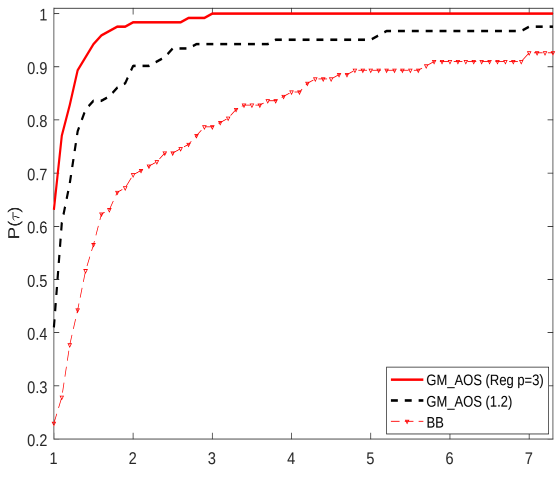


Fig. 3 N_g(CUTEr145).

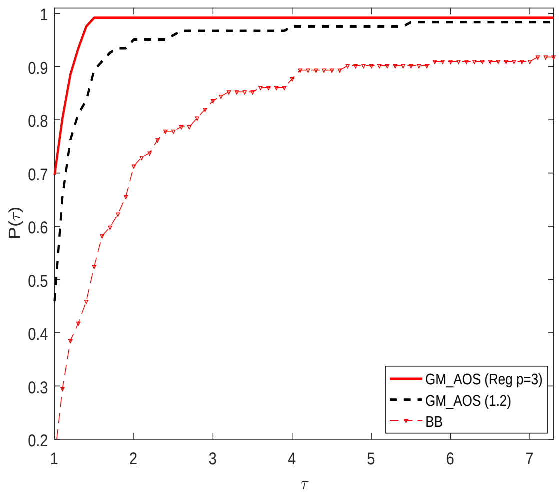


Fig. 4 T_{cpu}(CUTEr145).

The performance profiles introduced by Dolan and Moré [38] were used to display the performance of these methods. In the following figures, "N_{iter}", "N_f", "N_g" and "T_{cpu}" represent the number of iterations, the number of function evaluations, the number of gradient evaluations and CPU time (s), respectively.

The numerical experiments are divided into the following four groups.

In the first group of the numerical experiments, we compare the performance of GM_AOS (Reg p=3) with that of GM_AOS (1.2) [13] and the BB method on the test set CUTEr145. Figs. 1-4 present the performance profiles on the test set CUTEr145. As shown in Figs. 1-4, we can observe that GM_AOS (Reg p=3) performs better than GM_AOS (1.2) and is superior very much to the BB method, and GM_AOS (1.2) outperforms the BB method. The first group of the numerical experiments indicates that the approximately optimal stepsizes described in Section 2 are quite efficient.

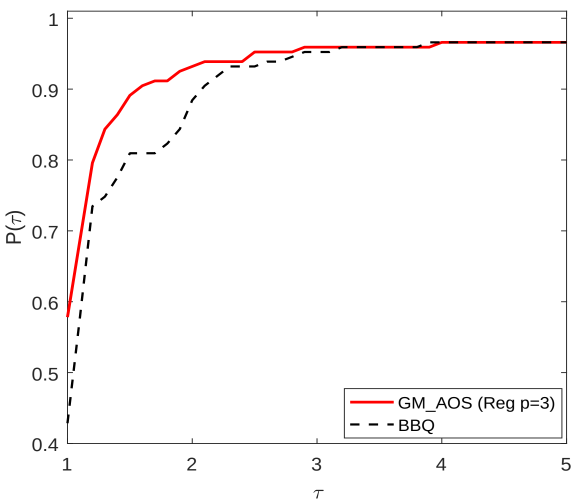


Fig. 5 N_{iter}

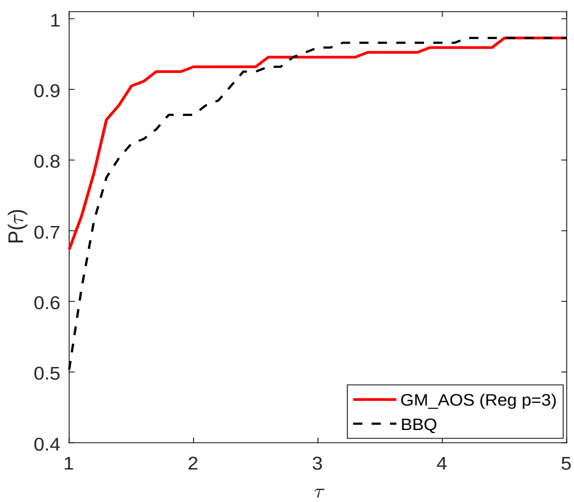


Fig. 6 N_f

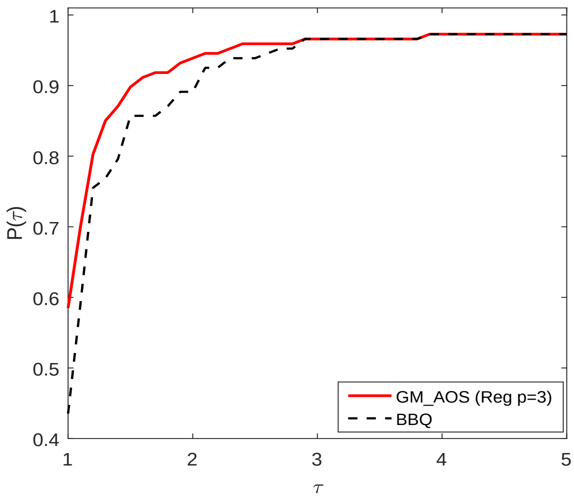


Fig. 7 N_g

In the second group of the numerical experiments, we compare the numerical performance of GM_AOS (Reg p=3) with that of the HDL method [16] on the same 147 test problems from the CUTEst library, which can be found in Dai's homepage. We do not compare the performance about the running time due to the fact that the HDL method was implemented by Matlab code and GM_AOS (Reg p=3) was implemented by C code. As shown in Fig. 5, 6 and 7, we can observed that GM_AOS (Reg p=3) is superior to the HDL method in term of the number of iteration, the number of function evaluation and the number of gradient evaluation, while the HDL method has been regarded as an import advance of gradient method.

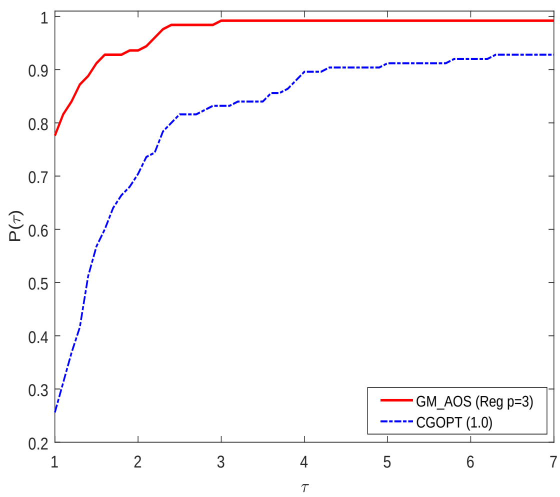


Fig. 8 N_f(CUTEr145)

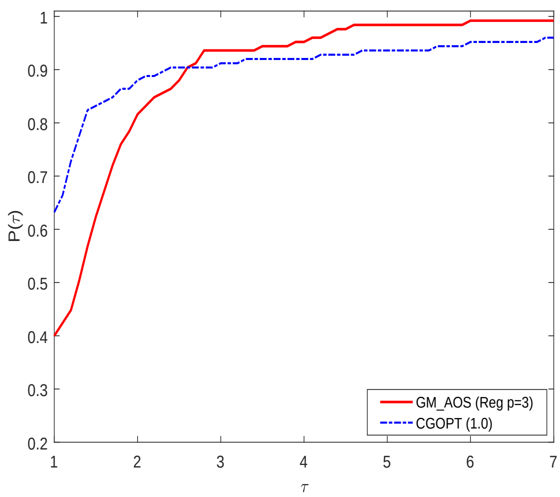


Fig. 9 N_g(CUTEr145)

In the third group of the numerical experiments, we compare the performance of GM_AOS (Reg p=3) with that of CGOPT (1.0) on the two test sets CUTEr145 and Andr80. Figs. 8-11 present the performance profiles on the test set CUTEr145. As shown in Fig. 8, we see that GM_AOS (Reg p=3) performs much better CGOPT (1.0) in term of N_f, since GM_AOS (Reg p=3) solves successfully about 79% test problems

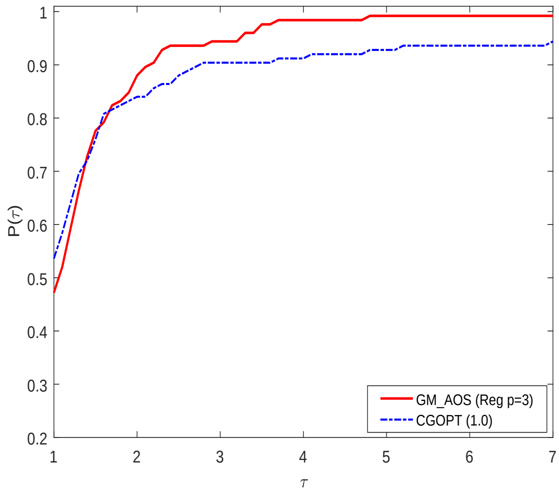


Fig. 10 $N_f + 3N_g$(CUTEr145).

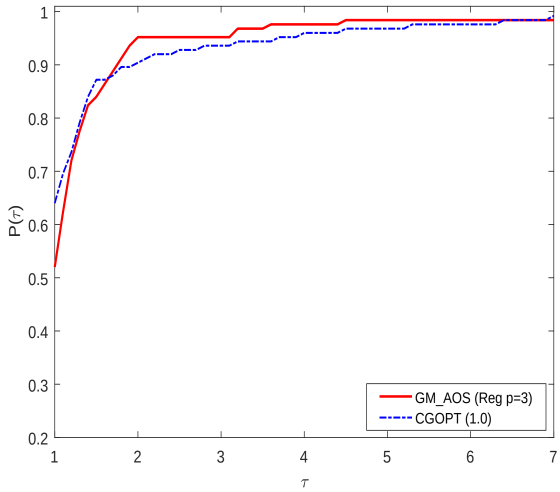


Fig. 11 T_{cpu}(CUTEr145).

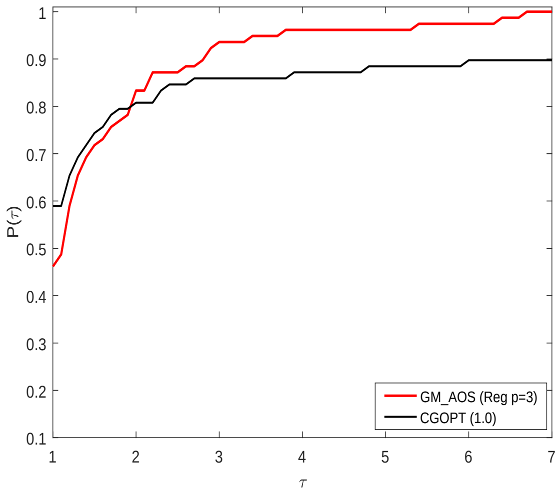


Fig. 12 N_{iter}(Andr80)

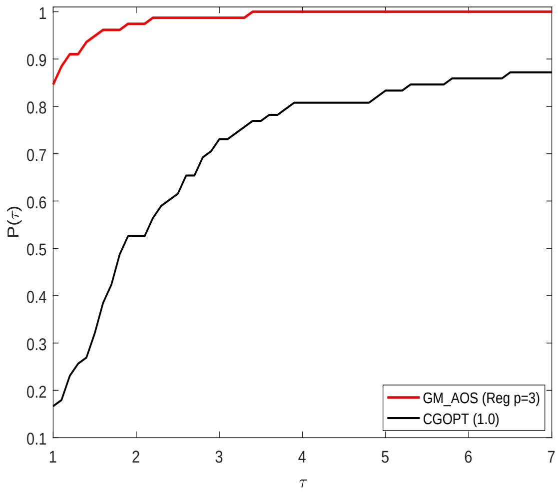
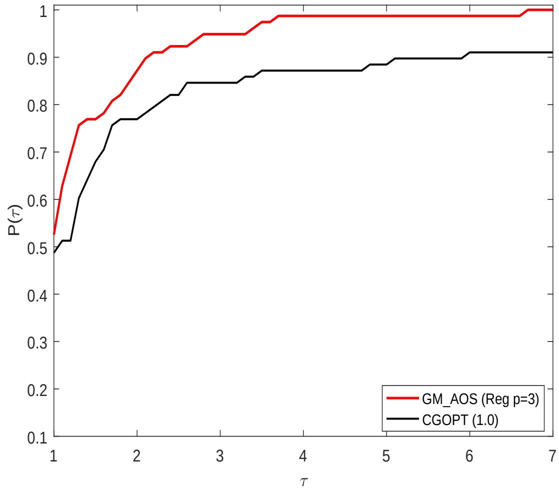


Fig. 13 N_f(Andr80)



5

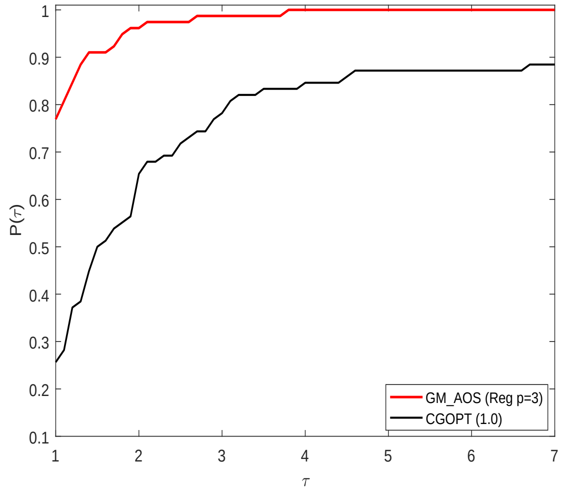


Fig. 14 N_g(Andr80).

Fig. 15 T_{cpu}(Andr80).

with the least function evaluations, while the percentage of CGOPT (1.0) is only about 38%. Fig. 9 indicates that GM_AOS (Reg p=3) is at a disadvantage over CGOPT (1.0) in term of N_g, and Fig. 10 shows that GM_AOS (Reg p=3) outperforms slightly CGOPT (1.0) in term of $N_f + 3N_g$ [39]. We can observe from Fig. 11 that GM_AOS (Reg p=3) is as fast as CGOPT (1.0). Figs. 12-15 present the performance profiles on the test set Andr80. As shown in Figs. 12-15, we observe that GM_AOS (Reg p=3) illustrates huge advantage over CGOPT (1.0) on the test set Andr80. The third group of the numerical experiments indicates that GM_AOS (Reg p=3) is competitive to CGOPT (1.0) on the test set CUTEr145, and has a significant advantage over CGOPT (1.0) on the test set Andr80.

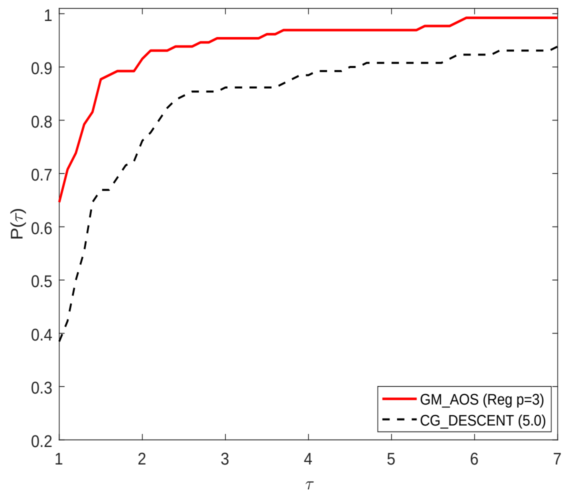


Fig. 16 N_f(CUTEr145)

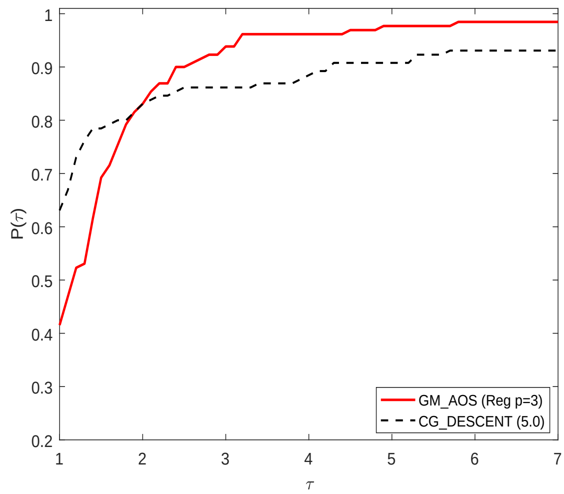


Fig. 17 N_g(CUTEr145)

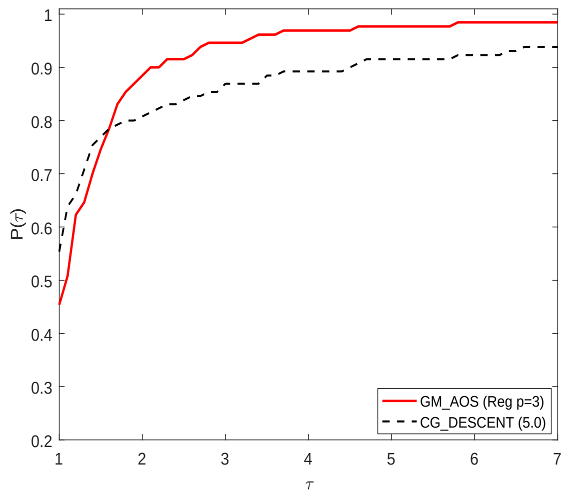


Fig. 18 $N_f + 3N_g$(CUTEr145)

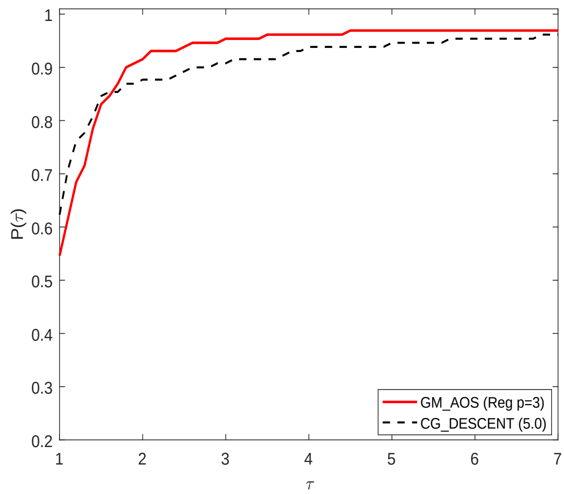


Fig. 19 T_{cpu}(CUTEr145)

In the fourth group of the numerical experiments, we compare the performance of GM_AOS (Reg p=3) with that of CG_DESCENT (5.0) on the two test sets CUTEr145 and Andr80. Figs. 16-19 present the

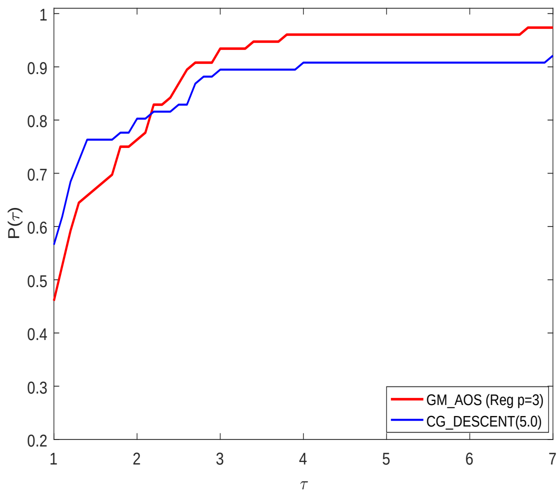


Fig. 20 N_{iter}(Andr80)

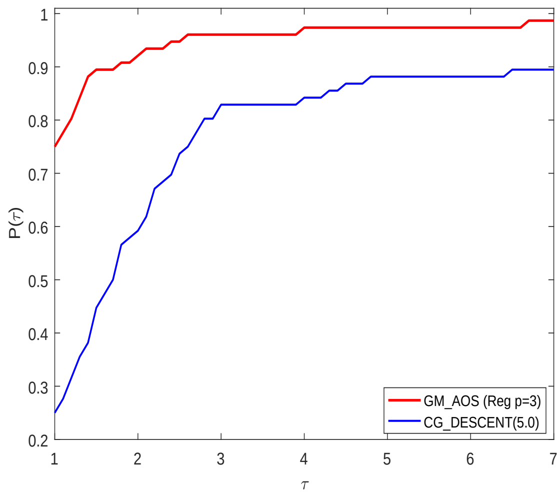


Fig. 21 N_f(Andr80)

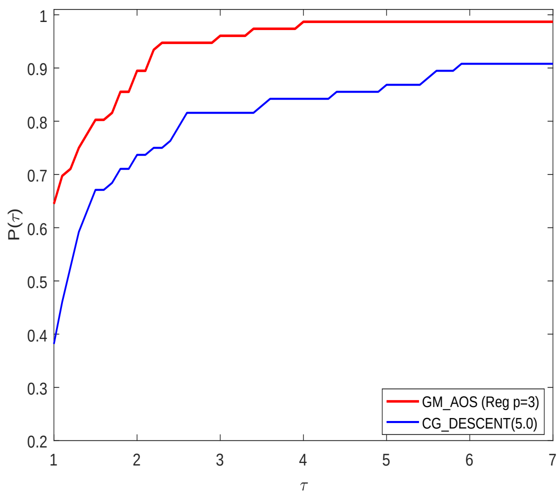


Fig. 22 N_g(Andr80)

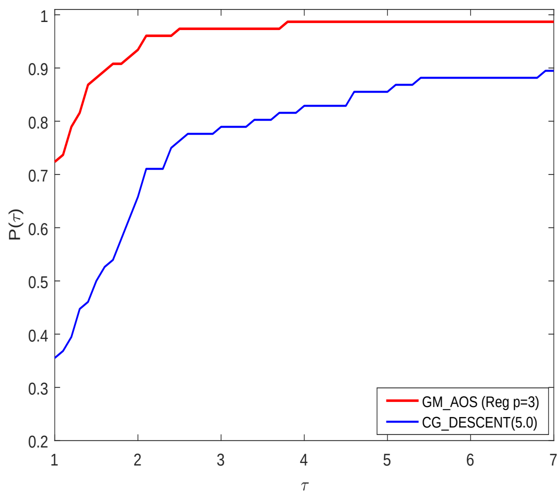


Fig. 23 T_{cpu}(Andr80)

performance profiles on the test set CUTEr145. As shown in Fig. 16, we see that GM_AOS (Reg p=3) performs better than CG_DESCENT (5.0) in term of N_f, since GM_AOS (Reg p=3) solves successfully about 65% test problems with the least function evaluations, while the percentage of CG_DESCENT (5.0) is only about 39%. Fig. 17 shows that GM_AOS (Reg p=3) is at a disadvantage over than CG_DESCENT (5.0) in term of N_g, and Fig. 18 indicates that GM_AOS (Reg p=3) outperforms slightly CG_DESCENT (5.0) in term of $N_f + 3N_g$ [39]. We can observe from Fig. 19 that GM_AOS (Reg p=3) is as fast as CG_DESCENT (5.0). Figs. 20-23 present the performance profiles on the test set Andr80. As shown in Figs. 20-22, we see that GM_AOS (Reg p=3) is at a little disadvantage over CG_DESCENT (5.0) in term of N_{iter}, and has a significant performance boost over CG_DESCENT (5.0) in term of N_f and N_g. We also can see that GM_AOS (Reg p=3) is faster much than CG_DESCENT (5.0). The fourth group of the numerical experiments indicates that GM_AOS (Reg p=3) is competitive to CG_DESCENT (5.0) on the test set CUTEr145, and has a significant advantage over CG_DESCENT (5.0) on the test set Andr80.

As for the reason that GM_AOS (Reg p=3) has so important improvement over CG_DESCENT (5.0) and CGOPT (1.0) on Andr80 and is only competitive to CG_DESCENT (5.0) and CGOPT (1.0) on CUTEr145, I think that it lies mainly in that most test problems in CUTEr145 is relatively difficult to solve compared to the test problems in Andr80. It seems that one can draw the following conclusion: Gradient methods with approximately optimal stepsize are sufficient for those test problems that are not very ill-conditioned.

Table 1. The number of test problems

Method	$N_{\text{linsear}} = 0$	$N_{\text{linsear}} \leq 1$	$N_{\text{linsear}} \leq 2$	$N_{\text{linsear}} \leq 3$	total problems
BB	41	46	48	50	145(CUTEr145)
GM_AOS (Reg p=3)	**68**	81	85	**90**	145(CUTEr145)

As for the reasons for the surprising numerical performance of GM_AOS (Reg p=3), we think that it lies in two aspects: (i)The approximately optimal stepsizes are generated by the approximation models including regularization models and quadratic models at the current iterate x_k. Since these approximation models possess rich second or higher order infomation of the objection function at the current iterate x_k, the resulted approximately optimal stepsize is integrated into rich second or higher order infomation properly and thus is very efficient. (ii)The approximately optimal stepsize can readily satisfy Zhang-Hager line search directly in most cases compared to other stepsizes in gradient method, which implies that it requires less much function evaluations and thus save much computational cost. This can be observed in Figs. 2, 8, 13, 16 and 21. Some statistical results can be seen in Table 1, where N_{linsear} denotes the times that the stepsize is updated by (30) during all iterations of solving a test problem. $N_{\text{linsear}} = 0$ indicates the initial stepsize (approximately optimal stepsize or BB stepsize) satisfies (27) directly at all iterations and thus **Zhang-Hager line search is not invoked at all**. As shown in Table 1, we can see that there are 68 (out of 145) problems for which Zhang-Hager line search is not invoked at all during the solving process, while the number for the BB method is only 41, and there are 90 (out of 145) problems for each of which the times that Zhang-Hager line search is invoked is less than or equal to 3, while the number for the BB method is only 50. It is observed from the Table 1 that the approximately optimal stepsizes described in Section 2 satisfy (27) in most cases and thus the proposed method requires less much function evaluations.

6 Conclusion and discussion

In this paper, we present an efficient gradient method with approximately optimal stepsizes for unconstrained optimization. In the proposed method, some approximation models including regularization models and quadratic models are exploited carefully to derive approximately optimal stepsizes. The convergence of the proposed methods is analyzed. Extensive numerical results indicates that the proposed method GM_AOS (Reg p=3) is very prosimising.

Due to the surprising numerical performance, gradient methods with approximately optimal stepsizes can become strong candidates for large scale unconstrained optimization and has potential in constrained optimization and some fields such as machine learning.

Though gradient methods with approximately optimal stepsize is surprisingly efficient, there are still some questions under investigation:

(i)Like the BB method, it is very challenging to explain that gradient methods with approximately optimal stepsizes converge so fast in theory. Does gradient method with approximately optimal stepsize based on quadratic approximation model (17) possess Q-linear convergence for convex quadratic minimization? If yes, what conditions should be imposed on the distance $\|B_k - A\|$? Here A is the Hessian matrix for strictly convex quadratic function.

(ii)Can the type of gradient method with approximately optimal stepsize possess local R-linear convergence or better convergence rate when it is applied to general unconstrained optimization?

Acknowledgements We would like to thank the associate editor and two anonymous referees for their valuable comments. We also would like to thank Professors Hager W. W. and Zhang, H. C. for their C code of CG_DESCENT, and thank Professor Dai Y. H. and Dr. Kou C. X. for their C code of CGOPT (1.0). This research is supported by National Science Foundation of China (No. 11901561) and Guizhou Provincial Science and Technology Projects (No. QKHJC-ZK[2022]YB084).

References

1. Cauchy A. Méthode générale pour la résolution des systéms déquations simultanées. Comp. Rend. Sci. Paris. 1847, 25: 46-89.
2. Akaike H. On a successive transformation of probability distribution and its application to the analysis of the optimum gradient method. Ann. Inst. Statist. Math. 1959, 11: 1-17.
3. Barzilai J., Borwein J. M. Two-point step size gradient methods. IMA J. Numer. Anal. 1988, 8(1): 141-148.
4. Raydan M. On the Barzilai and Borwein choice of steplength for the gradient method. IMA J. Numer. Anal. 1993, 13(3): 321-326.
5. Dai Y. H., Liao L. Z. R-linear convergence of the Barzilai and Borwein gradient method. IMA J. Numer. Anal. 2002, 22(1): 1-10.
6. Li D. W., Sun R. Y. On a faster R-linear convergence rate of the Barzilai-Borwein method. 2020, arXiv preprint arXiv: 2101.00205.
7. Dai Y. H., Zhang H. C. An adaptive two-point stepsize gradient algorithm. Numer. Algorithms 2001, 27(4): 377-385.
8. Raydan M. The Barzilai and Borwein gradient method for the large scale unconstrained minimization problem. SIAM J. Optim. 1997, 7: 26-33.
9. Grippo L., Lamparillo F., Lucidi S. A nonmonotone line search technique for Newton's method. SIAM J. Numer. Anal. 1986, 23: 707-716.
10. Dai Y. H., Yuan J. Y., Yuan Y. X. Modified two-point stepsize gradient methods for unconstrained optimization. Comput. Optim. Appl. 2002. 22(1): 103-109.
11. Liu Z. X., Liu H. W., Dong X. L. An efficient gradient method with approximate optimal stepsize for the strictly convex quadratic minimization problem. Optimization 2018, 67(3): 427-440.
12. Liu Z. X., Liu H. W. An efficient gradient method with approximate optimal stepsize for large-scale unconstrained optimization. Numer. Algorithms 2018, 78(1): 21-39.

13. Liu Z. X., Liu H. W. Several efficient gradient methods with approximate optimal stepsizes for large scale unconstrained optimization. J. Comput. Appl. Math. 2018, 328: 400-413.
14. Biglari F., Solimanpur M. Scaling on the spectral gradient method. J. Optim. Theory Appl. 2013, 158(2): 626-635.
15. Yuan Y. X. A new stepsize for the steepest descent method. J. Comput. Math. 2006, 24: 149-156.
16. Huang Y. K., Dai Y. H., Liu X. W. Equipping Barzilai-Borwein method with two dimensional quadratic termination property. SIAM J. Optim. 2021, 31(4): 3068–3096.
17. Dai Y. H., Hager W. W., Schittkowski K., et al. The cyclic Barzilai-Borwein method for unconstrained optimization. IMA J. Numer. Anal. 2006, 26(3): 604-627.
18. Xiao Y. H., Wang Q. Y., Wang D., et al. Notes on the Dai-Yuan-Yuan modified spectral gradient method. J. Comput. Appl. Math. 2010, 234(10): 2986-2992.
19. Nosratipour H., Fard O. S., Borzabadi A. H. An adaptive nonmonotone global Barzilai-Borwein gradient method for unconstrained optimization. Optimization 2017, 66(4): 641-655.
20. Miladinović M., Stanimirović P., Miljković S. Scalar correction method for solving large scale unconstrained minimization problems. J. Optim. Theory Appl. 2011, 151(2): 304-320.
21. Dai Y. H., Kou C. X. A nonlinear conjugate gradient algorithm with an optimal property and an improved Wolfe line search. SIAM J. Optim. 2013, 23(1): 296-320.
22. Hager W. W., Zhang H. C. A new conjugate gradient method with guaranteed descent and an efficient line search. SIAM J. Optim. 2005, 16(1): 170-192.
23. Gould N. I. M., Orban D., Toint Ph. L. CUTEr and SifDec: A constrained and unconstrained testing environment, revisited. ACM Trans. Math. Softw. 2003, 29(4): 373-394.
24. Andrei, N. An unconstrained optimization test functions collection. Adv. Model. Optim. 2008, 10: 147-161.
25. Sun W. Y., Xu. D. A filter-trust-region method based on conic model for unconstrained optimization. Sci. Sin. Math. 2012, 55: 527-543.
26. Sun W. Y. Optimization methods for non-quadratic model. Asia-Pac. J. Oper. Res. 1996, 13: 43-63
27. Cartis C., Gould N. I., Toint P. L. Adaptive cubic regularisation methods for unconstrained optimization. Part I: motivation, convergence and numerical results. Math. Program. 2011, 127(2): 245-295.
28. Cartis C., Gould N. I., Toint P. L. Adaptive cubic regularisation methods for unconstrained optimization. Part II: worst-case function-and derivative-evaluation complexity. Math. Program. 2011, 130(2): 295-319.
29. Bianconcini T., Liuzzi G., Morini B. On the use of iterative methods in cubic regularization for unconstrained optimization. Comput. Optim. Appl. 2015, 60(1): 35-57.
30. Toint P L. A non-monotone trust-region algorithm for nonlinear optimization subject to convex constraints. Math. Program. 1997, 77(3): 69-94.
31. Bianconcini T., Sciandrone M. Q. A cubic regularization algorithm for unconstrained optimization using line search and nonmonotone techniques. Optim. Methods Softw. 2016, 31: 1008-1035.
32. Zhang J. Z., Deng N. Y., Chen L. H. New quasi-Newton equation and related methods for unconstrained optimization. J. Optim. Theory Appl. 1999, 102(1): 147-167.
33. Cartis C., Gould N. I. M., Toint P. L. Adaptive cubic regularisation methods for unconstrained optimization. Part i: motivation, convergence and numerical results. Math. Program. 2011, 127: 245-295.
34. Gould N. I. M, Porcelli M. Updating the regularization parameter in the adaptive cubic regularization algorithm. Comput. Optim. Appl. 2012, 53: 1-22.
35. Luengo F., Raydan M. Gradient method with dynamical retards for large-scale optimization problems. Electron. Trans. Numer. Anal. 2003, 16(539): 186–193.
36. Zhang H. C., Hager W. W. A nonmonotone line search technique and its application to unconstrained optimization. SIAM J. Optim. 2004, 14(4): 1043-1056.
37. Birgin E. G., Martínez J. M., Raydan M. Nonmonotone spectral projected gradient methods for convex sets. SIAM J. Optim. 2000, 10: 1196-1211.

38. Dolan E. D., Moré J. J. Benchmarking optimization software with performance profiles. Math. Program. 2002, 91: 201-213.
39. Hager W. W., Zhang H. C. Algorithm 851: CG_DESCENT, a conjugate gradient method with guaranteed descent. ACM Trans. Math. Softw. 2006, 32(1): 113-137.