Математички Билтен
47 (LXXIII) 2 (2023), 147–164

UDC: 519.853:519.856
DOI:10.37560/matbil23472147n

SCALE INVARIANT STOCHASTIC GRADIENT METHOD WITH MOMENTUM

FILIP NIKOLOVSKI [1] AND IRENA STOJKOVSKA [2]

Abstract. Optimization in noisy environments arises frequently in applications. Solving this problem quickly, efficiently, and accurately is therefore of great importance. The stochastic gradient descent (SGD) method has proven to be a fundamental and an effective tool which is flexible enough to allow modifications for improving its convergence properties. In this paper we propose a new algorithm for solving an unconstrained optimization problems in noisy environments which combines the SGD with a modified momentum term using a two-point step size estimation in the Barzilai-Borwein (BB) framework. We perform a high probability analysis for the proposed algorithm and we establish its convergence under the standard assumptions. Numerical experiments demonstrate a promising behavior of the proposed method compared to the "vanilla" SGD with momentum in noise-free and in noisy environment when the objective function is scaled.

1. INTRODUCTION

We consider the unconstrained optimization problem:

$$\min_{x\in\mathbb{R}^n} f(x), \tag{1.1}$$

where the function $f : \mathbb{R}^n \to \mathbb{R}$ is at least twice continuously differentiable and it is bounded below; we denote its infimum by f^*, and by $\nabla f(x)$ and $\nabla^2 f(x)$ the gradient and the Hessian of f, respectively. We consider the case when only noisy estimates of the gradient of f are available at every point x i.e. the stochastic gradient $g(x, \xi)$ where ξ is a random noise.

One of the simplest stochastic optimization methods for solving (1.1) when only noisy measurements of the gradient are known is the *stochastic gradient descent* (SGD). Starting from an initial approximation x_1 of the solution, this method

2010 *Mathematics Subject Classification.* Primary: 90C30 (Nonlinear programming), 90C15 (Stochastic programming).

Key words and phrases. numerical optimization, stochastic gradient method, Barzilai-Borwein method, momentum method, scale invariance, high probablity convergence.

generates a recursive sequence of iterates by:

$$x_{k+1} = x_k - \mu g(x_k, \xi_k), \tag{1.2}$$

for $k = 1, 2, \ldots$, where $\mu > 0$ is a learning rate. In the classic version of the stochastic gradient method by Robbins and Monro [19], called *stochastic approximation* (SA), the constant μ is replaced by a sequence μ_k of suitably chosen constants called *step sizes*, and the convergence analysis of the method relies on two conditions for step sizes μ_k:

$$\sum_{k=1}^{\infty} \mu_k = \infty \quad \text{and} \quad \sum_{k=1}^{\infty} \mu_k^2 < \infty. \tag{1.3}$$

The conditions (1.3) ensure the convergence of the method, but SA is rather slow in practice which motivated researchers to propose accelerations and modifications of the SA method. One of the successful modifications is the popular AdaGrad method [6] which uses a step size sequence that does not satisfy the second condition in (1.3), but the method still converges in convex setting.

Another way to accelerate the SGD method is to add a "momentum" term in the gradient direction, as first proposed by Polyak [17]:

$$\begin{aligned} d_k &= \gamma d_{k-1} + \mu g(x_k, \xi_k) \\ x_{k+1} &= x_k - d_k, \end{aligned} \tag{1.4}$$

for $k = 1, 2, \ldots$, with $d_0 = 0$ and some constants $\mu > 0$ and $\gamma \in (0, 1)$. This approach is called *(classical) momentum method* or *heavy ball method* since the first equation in (1.4) ensures that some "momentum" from the previous iterations is carried over to the current iteration and is influencing the direction of the method in the current iteration, thus resembling a ball rolling down an inclined plane. This approach to acceleration of the gradient method has given rise to a plethora of highly efficient methods: Nesterov's Accelerated Gradient Descent (NAG) [13], RMSProp [20], and Adam [8] among others. In some of the momentum methods there are specifications of the learning rates, such as substituting the constant step sizes μ in (1.4) with a non-increasing sequence μ_k, [10]. The momentum methods have been successfully applied to training deep neural networks and have demonstrated to be well-suited for the task.

Other common approach to ensure better convergence of the gradient methods is when constants such as μ in (1.2), γ and μ in (1.4) are *tuned* based on the scales of the terms used in the recursive equations. But these tuned quantities only become available at the end of the learning process, [7]. This might be a problem with online algorithms that must assume some prior knowledge about the range of parameters and gradients is given, which allows the algorithm to tune its parameters appropriately. The *scale-invariant* algorithms aim to get rid of these range factors by a prior bound on the norms, [6, 16], or a prior bound on all future gradients [15]. For instance, the scale-invariance for linear regression models is studied and successfully resolved by Kempka et al., [7].

In this paper we propose a new approach to avoid tuning the parameters while ensuring the scale-invariance of the method. The new method is a momentum

method that incorporates ideas from the adaptive SGD with momentum [10] and the Barzilai-Borwen (BB) method [1]. We also perform a high-probability convergence analysis of the behavior of the proposed method in a noisy environment under standard assumptions. The performance of the proposed method is numerically tested and compared to the performance of the "vanilla" stochastic gradient descent method with momentum on a set of quadratic and non-quadratic functions, in both noisy and noise-free environment. Numerical results show a promising behavior of the proposed method when the objective function is scaled.

The paper is organized as follows. In Section 2, the Barzilai-Borwen (BB) method for a deterministic unconstrained optimization problem is described with its advantages over the gradient methods and the quasi-Newton methods. A new momentum method with BB information is proposed in Section 3; its scale-invariance in a noise-free environment is proved. In Section 4, the high-probability convergence analysis of the proposed method is performed and numerical results are presented in Section 5. In Section 6, some conclusion are given.

2. THE BARZILAI-BORWEIN METHOD

When solving the deterministic unconstrained optimization problem (1.1) in noise-free environment (i.e. when $\xi = 0$), a notable improvement of the gradient methods can be achieved by including second order derivatives of the objective function. However, this is generally not always possible and is much more computationally expensive. The *quasi-Newton methods* give one possible way to avoid calculating second order derivatives and still use second order information by taking an approximation B_k of the true Hessian $\nabla^2 f(x_k)$. Thus, the quasi-Newton methods take the iteration of the form:

$$x_{k+1} = x_k - \eta_k B_k^{-1} \nabla f(x_k), \tag{2.1}$$

where $\eta_k > 0$ is a step size. The approximate Hessian B_k needs to satisfy the *secant equation*:

$$B_k s_k = y_k, \tag{2.2}$$

where $s_k = x_k - x_{k-1}$ and $y_k = \nabla f(x_k) - \nabla f(x_{k-1})$, for $k \geq 1$. But, in practice, when n is large, it is time consuming to compute the B_k.

The *Barzilai and Borwein (BB) method* takes a different approach in using a second order information, [1]. Unlike the quasi-Newton methods which at every iteration require computation of full-rank matrices that satisfy the secant equation (2.2), the BB method uses a diagonal matrix of the form αI, where I is the $n \times n$ identity matrix and $\alpha \in \mathbb{R}$; then the secant equation (2.2) is solved *approximately* for a matrix of the specified form. In this context, the BB iteration is of the form:

$$x_{k+1} = x_k - \alpha_k \nabla f(x_k), \tag{2.3}$$

with the step size α_k at the k-the iteration chosen as a solution of the problem

$$\min_{\alpha} \left\| \frac{1}{\alpha} s_k - y_k \right\|^2, \tag{2.4}$$

where $\|\cdot\|$ is the Euclidean norm. When $s_k^T y_k > 0$, the solution of (2.4) gives the step size α_k of the form:

$$\alpha_k = \alpha_k^{BB1} = \frac{s_k^T s_k}{s_k^T y_k}. \tag{2.5}$$

An alternative way to obtain α_k is as a solution of the problem

$$\min_{\alpha} \|s_k - \alpha y_k\|^2, \tag{2.6}$$

which gives a step size α_k of the form:

$$\alpha_k = \alpha_k^{BB2} = \frac{s_k^T y_k}{y_k^T y_k}. \tag{2.7}$$

An obvious advantage of the BB method is the fact that it incorporates information about the geometry of the objective function into the step size at a very low additional computational cost. As discussed in [2], the performance of the BB method is *R*-superlinear in the case of a two or three dimensional quadratic objective, while it is *R*-linear for higher dimensional quadratic and for non-quadratic functions; nevertheless, numerical experiments show that both variants of the BB method, (2.5) and (2.7), significantly improve the performance of the gradient method, [2].

When only noisy measurements of the gradient are known, the *stochastic variants* of the both the quasi-Newton and BB methods (obtained by simple substitution of the gradient with a noisy gradient) are not recommended. This is because both methods can produce large step sizes near a solution which may lead to divergence. There are modifications of quasi-Newton methods, [9, 25], and BB methods, [23, 26], for optimization problem in noisy environment (1.1), that overcome these issues by including additional strategies.

3. New momentum method

As we pointed before, momentum methods (1.4) and their variants are already successfully used for solving stochastic problems, such as in training deep neural networks. But there is an obvious room for improvement when solving the general unconstrained optimization problem in noisy environment (1.1) since there is a possibility that the momentum method is drawn to a local minimum, especially when (as frequently is the case) the constant μ in (1.4) is chosen as a decreasing sequence $\{\mu_k\}$; see [10]. On the other hand, having non-constant parameter μ_k gives method a chance for self-correction.

We propose adding the BB information in each step of the momentum method in [10], which gives the following *Stochastic Gradient Method with Momentum and Barzilai-Borwein Information (SGMBB)* for solving the unconstrained optimization problem in noisy environment (1.1):

$$\begin{aligned} d_k &= \gamma d_{k-1} + \mu_k \alpha_k g(x_k, \xi_k) \\ x_{k+1} &= x_k - d_k, \end{aligned} \tag{3.1}$$

for $k = 1, 2, \ldots$, with $d_0 = 0$, $\alpha_1 = 1/\|g_1\|$, where $g_1 = g(x_1, \xi_1)$, $\gamma \in (0, 1)$, $\{\mu_k\}$ is a non-increasing positive sequence and $\alpha_k > 0$ is defined as

$$\alpha_k = \frac{s_k^T s_k}{s_k^T y_k},$$
where
$$s_k = x_k - x_{k-1}, \quad y_k = g(x_k, \xi_{k-1}) - g(x_{k-1}, \xi_{k-1}). \tag{3.2}$$

We expect that the momentum method will improve its performance by adding an information of second order such as BB information (3.2). On the other hand, as we will show later, in a noise-free environment, the presence of α_k and the special form of its starting value, makes the search direction d_k invariant to the scaling of the objective function. Similar analysis of scale invariance in context of quasi-Newton methods can be found in [5].

Let us note that calculation of y_k in (3.2) uses the noise from $(k-1)$-th iteration for the both gradients $g(x_k, \xi_{k-1})$ and $g(x_{k-1}, \xi_{k-1})$. This might double the number of gradient calculations in experiments, but using the same sample set to calculate noisy gradients at different points in same iteration is more natural, and is already successfully tested in [3, 9, 21], indicated that the additional computation is well spent.

In practical implementation of the method, when the condition $\alpha_k > 0$ is not satisfied, one can always set a damping step $\alpha_k = \alpha_{k-1}$, frequently used in optimization algorithms.

The proposed method (3.1)-(3.2) is summarized in Algorithm 1.

Algorithm 1 SG with Momentum and BB Information (SGMBB)

1: **input:** $x_1 \in \mathbb{R}^n$, $N \in \mathbb{N}$, $\gamma \in (0, 1)$, $\{\mu_k\}_{k=1}^N$ non-increasing positive sequence
2: **sample:** $\{\xi_k\}_{k=1}^N$ independent with $\{\mu_k\}_{k=1}^N$
3: **set:** $d_0 = 0$, $g_1 = g(x_1, \xi_1)$, $\alpha_1 = 1/||g_1||$
4: **for** $k = 1, 2, \ldots, N$ **do**
5: $\quad d_k = \gamma d_{k-1} + \mu_k \alpha_k g(x_k, \xi_k)$
6: $\quad x_{k+1} = x_k - d_k$
7: $\quad$ **get:** s_{k+1}, y_{k+1}, and α_{k+1} as defined in (3.2)
8: **end for**

The assumption on the noise $\{\xi_k\}$ in line 2 of Algorithm 1 is technical and ensures that the delayed step sizes can avoid the possible deviation brought by the step sizes that include the current noise, as it is pointed out in [10].

3.1. **Scale invariance of the method in a noise-free environment.** The method we propose, given in Algorithm 1, as stated previously, makes the search direction d_k invariant to the scaling of the objective function when optimization is performed in noise-free environment, i.e. $\xi = 0$ and $g(x, \xi) = \nabla f(x)$ for all $x \in \mathbb{R}^n$.

Denote by $f^\omega = \omega f$, $\omega > 0$, the scaled variant of the objective function f. Then for the gradient of the scaled function we have $\nabla f^\omega = \omega \nabla f$. Denote by d_k^ω, α_k^ω and x_k^ω, the search direction, the BB step size and the iteration, respectively, for the scaled function. We will use the following abbreviations $\nabla f_k = \nabla f(x_k)$ and $\nabla f_k^\omega = \nabla f^\omega(x_k)$, for $k = 1, 2, ..., N$. We will prove, by induction on k, that the direction d_k is invariant to the scaling of the objective function i.e. $d_k^\omega = d_k$, for all $k = 1, 2, ..., N$. Consequently, for $x_1^\omega = x_1$, from Algorithm 1, line 6, we have $x_k^\omega = x_k$, for all $k = 1, 2, ..., N$, and the algorithm is scale-invariant.

Let $x_1^\omega = x_1$. From Algorithm 1, line 3, we have $d_0 = d_0^\omega = 0$. For $k = 1$ we have:

$$d_1^\omega = \mu_1 \frac{\nabla f_1^\omega}{\|\nabla f_1^\omega\|} = \mu_1 \frac{\omega \nabla f_1}{\|\omega \nabla f_1\|} = \mu_1 \frac{\nabla f_1}{\|\nabla f_1\|} = d_1,$$

which implies $x_2^\omega = x_2$. We assume $d_k^\omega = d_k$ up to some index k, and consequently $x_k^\omega = x_k$ up to $k+1$. Then $s_k^\omega = x_k^\omega - x_{k-1}^\omega = x_k - x_{k-1} = s_k$ up to $k+1$, and $y_k^\omega = \nabla f_k^\omega - \nabla f_{k-1}^\omega = \omega \nabla f_k - \omega \nabla f_{k-1} = \omega y_k$ up to $k+1$.

Then for the BB step α_k^ω up to $k+1$ we have:

$$\alpha_k^\omega = \frac{s_k^T s_k}{s_k^T (y_k^\omega)} = \frac{s_k^T s_k}{\omega s_k^T y_k} = \frac{1}{\omega} \alpha_k, \tag{3.3}$$

from where for the search directions d_{k+1}^ω and d_{k+1} we get:

$$d_{k+1}^\omega = d_k^\omega + \mu_{k+1} \alpha_{k+1}^\omega \nabla f_{k+1}^\omega = d_k + \mu_{k+1} \cdot \frac{1}{\omega} \alpha_{k+1} \cdot \omega \nabla f_{k+1} = d_{k+1}, \tag{3.4}$$

which proves by induction that $d_k^\omega = d_k$, for all $k = 1, 2, ..., N$, and consequently, if $x_1^\omega = x_1$, we have $x_k^\omega = x_k$, for all $k = 1, 2, ..., N$, and the algorithm is scale-invariant.

Note that the direction d_k will be invariant to the scaling of the objective function if we have used a stochastic variant of BB step of the form (2.7), instead of the form (2.5), as we did in Algorithm 1.

4. Convergence analysis

When solving the problem (1.1), the presence of noise makes gradients and iterates generated by Algorithm 1 stochastic variables. In such setting, the convergence analysis lies on martingales inequalities. In this section, we perform a high probability convergence analysis, which can capture the convergence behavior of the algorithm when only a small number of runs of the algorithm are performed, as it is typical in modern machine learning applications, [11].

4.1. **Assumptions.** Let $\{x_k\}$ be a sequence generated by Algorithm 1. Denote by $\mathcal{F}_k$ the σ-algebra generated by $x_1, x_2, ..., x_k$. We denote by $\mathbb{E}[\cdot \mid \mathcal{F}_k]$ the conditional expectation with respect to the σ-algebra $\mathcal{F}_k$. Let ξ_k denote the noise in k-th iteration. We make the following assumptions:

(A1) The objective function f is twice continuously differentiable, bounded below with $f^* = \inf_x f(x)$ and its gradient ∇f is Lipschitz continuous i.e. $\exists L > 0, \|\nabla f(x) - \nabla f(y)\| \leqslant L\|x - y\|, \forall x, y \in \mathbb{R}^n$.

(A2) The Hessian $\nabla^2 f$ is uniformly bounded i.e. there exist $0 < m \leqslant M$ such that $\|y\|^2/M \leqslant y^T \nabla^2 f(x) y \leqslant \|y\|^2/m$, $\forall x, y \in \mathbb{R}^n$.
(A3) The noisy gradient g is unbiased estimate of ∇f i.e. $\mathbb{E}[g(x_k, \xi_k) \mid \mathcal{F}_k] = \nabla f(x_k)$, for all k.
(A4) The noise is *sub-Gaussian*, i.e. $\mathbb{E}\left[\exp\left(\|\nabla f(x_k) - g(x_k, \xi_k)\|^2/\sigma^2\right) \mid \mathcal{F}_k\right] \leqslant \exp(1)$, for all k and some $\sigma > 0$.

The Lipschitz continuity of the gradient ∇f in assumption (A1) is necessary for convergence of the gradients to zero, [10]. Assumption (A1) also implies, [14]:

$$|f(y) - f(x) - \nabla f(x)^T (y - x)| \leqslant \frac{L}{2}\|y - x\|^2. \tag{4.1}$$

The uniform boundedness of the Hessian $\nabla^2 f$ in assumption (A2) in the case of an *additive gradient noise*:

$$g(x, \xi) = \nabla f(x) + \xi, \tag{4.2}$$

implies boundedness of BB steps α_k defined with (3.2) i.e.

$$m \leqslant \alpha_k \leqslant M, \text{for all } k. \tag{4.3}$$

Indeed, when the gradient noise is additive (4.2), y_k defined in (3.2) becomes:

$$y_k = g(x_k, \xi_{k-1}) - g(x_{k-1}, \xi_{k-1}) = \nabla f(x_k) - \nabla f(x_{k-1}), \tag{4.4}$$

and the proof of (4.3) follows directly from the same reasoning in the deterministic case, see [2] or [22]. Statement (4.3) makes sense as it has the added property of keeping the BB steps (3.2) from becoming too large or too small. Similar approach in noisy environments has frequently been taken, see for example [24, 26].

In the case of an additive gradient noise (4.2), assumption (A3) implies that the observation noise $\{\xi_k, \mathcal{F}_{k+1}\}$ is a martingale difference sequence i.e.

$$\mathbb{E}[\|\xi_k\|] < +\infty \quad \text{and} \quad \mathbb{E}[\xi_k \mid \mathcal{F}_k] = 0. \tag{4.5}$$

Moreover, assumption (A4) and Jensen's inequality imply

$$\mathbb{E}[\|\xi_k\|^2 \mid \mathcal{F}_k] \leqslant \sigma^2, \tag{4.6}$$

which establishes σ from assumption (A4) as *noise level* of the gradient noise. Conditions (4.5) and (4.6) are commonly used in stochastic optimization methods. Assumption (A4) also intuitively implies that the tails of the noise distribution are dominated by tales of a Gaussian distribution, [10].

4.2. **High probability analysis.** Here we show results that guarantee the convergence of the proposed method in high probability. Our high probability convergence analysis is inspired by [10, 11, 27]. For simplicity, we write $f_k = f(x_k)$, $\nabla f_k = \nabla f(x_k)$ and $g_k = g(x_k, \xi_k)$.

Our high probability analysis lies on the following result for martingale difference sequences.

Lemma 1. *Assume that* $\{Y_k, \mathcal{F}_{k+1}\}$ *is a martingale difference sequence and assume that* $\mathbb{E}[\exp(Y_k^2/Z_k^2) \mid \mathcal{F}_k] \leqslant \exp(1)$ *for all* $1 \leqslant k \leqslant N$, *where* Z_k *is* $\mathcal{F}_k$*-measurable sequence*

of random variables. Then for any fixed $\lambda > 0$ *and* $\delta \in (0,1)$*, with probability at least* $1-\delta$*, we have*

$$\sum_{k=1}^{N} Y_k \leqslant \frac{3}{4}\lambda \sum_{k=1}^{N} Z_k^2 + \frac{1}{\lambda}\ln\frac{1}{\delta}.$$

Proof. See the proof of Lemma 1 in [10]. □

The following lemma establishes some probability bounds needed for the main theorem.

Lemma 2. *Let assumptions (A1)–(A4) hold and let* $\{x_k\}$ *be a sequence generated by Algorithm 1. Let* $\delta \in (0,1)$*. Then in the case of an additive gradient noise (4.2), with probability at least* $1-\delta$*, the following holds:*

$$\sum_{k=1}^{N} \mu_k\alpha_k\|\nabla f_k\|^2 \leqslant C_1 \sum_{k=1}^{N} \mu_k^2\alpha_k^2\|g_k\|^2 + C_2,$$

where $C_1 = \frac{2L(3-\gamma)}{(1-\gamma)^3}$ *and* $C_2 = 2(f_1 - f^*) + \frac{3\mu_1 M(1-\gamma^N)^2\sigma^2}{(1-\gamma)^2}\cdot\ln\frac{1}{\delta}$.

Proof. The proof of this claim is similar to the proof of Lemma 2 in [10]. Here we additionally use the boundedness of BB steps (4.3).

Assumption (A1) implies the inequality (4.1) and by the definition of x_{k+1} in (3.1) we have:

$$\begin{aligned} f_{k+1} &\leqslant f_k + \nabla f_k^T(x_{k+1} - x_k) + \frac{L}{2}\|x_{k+1} - x_k\|^2 \\ &= f_k - \nabla f_k^T d_k + \frac{L}{2}\|d_k\|^2, \end{aligned}$$

for all $k = 1, \ldots, N$. By iterating the last inequality, we get:

$$\begin{aligned} f_{N+1} &\leqslant f_N - \nabla f_N^T d_N + \frac{L}{2}\|d_N\|^2 \\ &\leqslant f_{N-1} - \nabla f_{N-1}^T d_{N-1} + \frac{L}{2}\|d_{N-1}\|^2 - \nabla f_N^T d_N + \frac{L}{2}\|d_N\|^2 \\ &\vdots \\ &\leqslant f_1 - \sum_{k=1}^{N} \nabla f_k^T d_k + \sum_{k=1}^{N} \frac{L}{2}\|d_k\|^2. \end{aligned}$$

Since $f^* \leqslant f_{N+1}$, from the last inequality we get:

$$f^* - f_1 \leqslant -\sum_{k=1}^{N} \nabla f_k^T d_k + \sum_{k=1}^{N} \frac{L}{2}\|d_k\|^2. \tag{4.7}$$

Now we are going to find the upper bound of $-\nabla f_k^T d_k$ in the first term in (4.7). By the definition of d_k in (3.1) and assumption (A1), we have:

$$\begin{aligned} -\nabla f_k^T d_k &= -\nabla f_k^T(\gamma d_{k-1} + \mu_k\alpha_k g_k) \\ &= -\gamma\nabla f_k^T d_{k-1} - \mu_k\alpha_k\nabla f_k^T g_k \end{aligned}$$

$$
\begin{aligned}
&= -\gamma \nabla f_{k-1}^T d_{k-1} + \gamma \nabla f_{k-1}^T d_{k-1} - \gamma \nabla f_k^T d_{k-1} - \mu_k \alpha_k \nabla f_k^T g_k \\
&= -\gamma \nabla f_{k-1}^T d_{k-1} - \gamma (\nabla f_k - \nabla f_{k-1})^T d_{k-1} - \mu_k \alpha_k \nabla f_k^T g_k \\
&\leqslant -\gamma \nabla f_{k-1}^T d_{k-1} + \gamma \|\nabla f_k - \nabla f_{k-1}\| \|d_{k-1}\| - \mu_k \alpha_k \nabla f_k^T g_k \\
&\leqslant -\gamma \nabla f_{k-1}^T d_{k-1} + \gamma L \|d_{k-1}\|^2 - \mu_k \alpha_k \nabla f_k^T g_k
\end{aligned}
$$

By iterating the last inequality and using $d_0 = 0$, we get:

$$
\begin{aligned}
-\nabla f_k^T d_k &\leqslant -\gamma \nabla f_{k-1}^T d_{k-1} + \gamma L \|d_{k-1}\|^2 - \mu_k \alpha_k \nabla f_k^T g_k \\
&\leqslant -\gamma^2 \nabla f_{k-2}^T d_{k-2} + \gamma^2 L \|d_{k-2}\|^2 - \gamma \mu_{k-1} \alpha_{k-1} \nabla f_{k-1}^T g_{k-1} \\
&\quad + \gamma L \|d_{k-1}\|^2 - \mu_k \alpha_k \nabla f_k^T g_k \\
&\cdots \\
&\leqslant L \sum_{i=1}^{k-1} \gamma^{k-i} \|d_i\|^2 - \sum_{i=1}^{k} \gamma^{k-i} \mu_i \alpha_i \nabla f_i^T g_i
\end{aligned}
$$

If we substitute the last inequality in (4.7) and because of the additive gradient noise (4.2) i.e. $g_k = \nabla f_k + \xi_k$, we have:

$$
\begin{aligned}
f^* - f_1 &\leqslant L \sum_{k=1}^{N} \sum_{i=1}^{k-1} \gamma^{k-i} \|d_i\|^2 - \sum_{k=1}^{N} \sum_{i=1}^{k} \gamma^{k-i} \mu_i \alpha_i \nabla f_i^T g_i + \sum_{k=1}^{N} \frac{L}{2} \|d_k\|^2 \\
&= L \sum_{k=1}^{N} \sum_{i=1}^{k-1} \gamma^{k-i} \|d_i\|^2 - \sum_{k=1}^{N} \sum_{i=1}^{k} \gamma^{k-i} \mu_i \alpha_i \nabla f_i^T (\nabla f_i + \xi_i) + \sum_{k=1}^{N} \frac{L}{2} \|d_k\|^2 \\
&= L \sum_{k=1}^{N} \sum_{i=1}^{k-1} \gamma^{k-i} \|d_i\|^2 - \sum_{k=1}^{N} \sum_{i=1}^{k} \gamma^{k-i} \mu_i \alpha_i \|\nabla f_i\|^2 \\
&\quad - \sum_{k=1}^{N} \sum_{i=1}^{k} \gamma^{k-i} \mu_i \alpha_i \nabla f_i^T \xi_i + \sum_{k=1}^{N} \frac{L}{2} \|d_k\|^2,
\end{aligned}
\tag{4.8}
$$

For the first term on right side of (4.8) we have:

$$
\begin{aligned}
L \sum_{k=1}^{N} \sum_{i=1}^{k-1} \gamma^{k-i} \|d_i\|^2 &= L \sum_{k=1}^{N-1} \|d_k\|^2 \sum_{i=1}^{N-k} \gamma^i \leqslant L\gamma \sum_{k=1}^{N-1} \|d_k\|^2 \sum_{i=0}^{\infty} \gamma^i \\
&= \frac{L\gamma}{1-\gamma} \sum_{k=1}^{N-1} \|d_k\|^2 \leqslant \frac{L}{1-\gamma} \sum_{k=1}^{N} \|d_k\|^2.
\end{aligned}
\tag{4.9}
$$

For the second term on right side of (4.8) we have:

$$\begin{aligned} -\sum_{k=1}^{N}\sum_{i=1}^{k}\gamma^{k-i}\mu_i\alpha_i\|\nabla f_i\|^2 &= -\sum_{k=1}^{N}\mu_k\alpha_k\|\nabla f_k\|^2\sum_{i=0}^{N-k}\gamma^i \\ &= -\sum_{k=1}^{N}\frac{1-\gamma^{N-k+1}}{1-\gamma}\mu_k\alpha_k\|\nabla f_k\|^2 \\ &\leqslant -\sum_{k=1}^{N}\mu_k\alpha_k\|\nabla f_k\|^2 \end{aligned} \tag{4.10}$$

By the same reasoning, for the third term on right side of (4.8) we have:

$$-\sum_{k=1}^{N}\sum_{i=1}^{k}\gamma^{k-i}\mu_i\alpha_i\nabla f_i^T\xi_i = \sum_{k=1}^{N}\left(-\frac{1-\gamma^{N-k+1}}{1-\gamma}\mu_k\alpha_k\nabla f_k^T\xi_k\right) \triangleq \sum_{k=1}^{N}U_k, \tag{4.11}$$

where because of (4.5) i.e. $\mathbb{E}[\xi_k \mid \mathcal{F}_k] = 0$, for $U_k = -\frac{1-\gamma^{N-k+1}}{1-\gamma}\mu_k\alpha_k\nabla f_k^T\xi_k$ we have that $\mathbb{E}[U_k \mid \mathcal{F}_k] = 0$ and $\{U_k, \mathcal{F}_{k+1}\}$ is a martingale difference sequence. Denote by $V_k = \frac{(1-\gamma^{N-k+1})^2}{(1-\gamma)^2}\mu_k^2\alpha_k^2\|\nabla f_k\|^2\sigma^2$, then using the boundedness of the noise variance i.e. (4.6) we have:

$$\begin{aligned} \mathbb{E}\left[\exp\left(\frac{U_k^2}{V_k}\right)\middle|\mathcal{F}_k\right] &= \mathbb{E}\left[\exp\left(\frac{(\nabla f_k^T\xi_k)^2}{\|\nabla f_k\|^2\sigma^2}\right)\middle|\mathcal{F}_k\right] \\ &\leqslant \mathbb{E}\left[\exp\left(\frac{\|\nabla f_k\|^2\|\xi_k\|^2}{\|\nabla f_k\|^2\sigma^2}\right)\middle|\mathcal{F}_k\right] \\ &= \mathbb{E}\left[\exp\left(\frac{\|\xi_k\|^2}{\sigma^2}\right)\middle|\mathcal{F}_k\right] \leqslant \exp(1), \end{aligned}$$

and by Lemma 1 we get that for any fixed $\lambda > 0$ and $\delta \in (0,1)$, with probability at least $1-\delta$, we have:

$$\begin{aligned} \sum_{k=1}^{N}U_k &\leqslant \frac{3}{4}\lambda\sum_{k=1}^{N}V_k + \frac{1}{\lambda}\ln\frac{1}{\delta} \\ &= \frac{3}{4}\lambda\sum_{k=1}^{N}\frac{(1-\gamma^{N-k+1})^2}{(1-\gamma)^2}\mu_k^2\alpha_k^2\|\nabla f_k\|^2\sigma^2 + \frac{1}{\lambda}\ln\frac{1}{\delta} \\ &\leqslant \frac{3}{4}\lambda\sum_{k=1}^{N}\frac{(1-\gamma^{N})^2}{(1-\gamma)^2}\mu_k^2\alpha_k^2\|\nabla f_k\|^2\sigma^2 + \frac{1}{\lambda}\ln\frac{1}{\delta} \\ &\leqslant \frac{3\lambda\mu_1 M(1-\gamma^N)^2\sigma^2}{4(1-\gamma)^2}\sum_{k=1}^{N}\mu_k\alpha_k\|\nabla f_k\|^2 + \frac{1}{\lambda}\ln\frac{1}{\delta}, \end{aligned} \tag{4.12}$$

where the last inequality comes form $\mu_k \leqslant \mu_1$ and (4.3).

Now by the definition of d_k in (3.1) and by the convexity of $\|.\|^2$, we have:

$$\|d_k\|^2 = \|\gamma d_{k-1} + \mu_k\alpha_k g_k\|^2$$

$$
\begin{aligned}
&= \left\| \gamma d_{k-1} + (1-\gamma)\frac{\mu_k \alpha_k}{1-\gamma} g_k \right\|^2 \\
&\leqslant \gamma \|d_{k-1}\|^2 + (1-\gamma)\frac{\mu_k^2 \alpha_k^2}{(1-\gamma)^2} \|g_k\|^2 \\
&= \gamma \|d_{k-1}\|^2 + \frac{\mu_k^2 \alpha_k^2}{1-\gamma} \|g_k\|^2,
\end{aligned}
$$

and summing the last inequality over $k = 1, 2, ..., N$, having in mind that $d_0 = 0$, we get:

$$
\begin{aligned}
\sum_{k=1}^{N} \|d_k\|^2 &\leqslant \gamma \sum_{k=1}^{N} \|d_{k-1}\|^2 + \frac{1}{1-\gamma} \sum_{k=1}^{N} \mu_k^2 \alpha_k^2 \|g_k\|^2 \\
&\leqslant \gamma \sum_{k=1}^{N} \|d_k\|^2 + \frac{1}{1-\gamma} \sum_{k=1}^{N} \mu_k^2 \alpha_k^2 \|g_k\|^2.
\end{aligned}
$$

From the last inequality we get the following bound:

$$
\sum_{k=1}^{N} \|d_k\|^2 \leqslant \frac{1}{(1-\gamma)^2} \sum_{k=1}^{N} \mu_k^2 \alpha_k^2 \|g_k\|^2. \tag{4.13}
$$

Now, from (4.8)-(4.13), for $\lambda = \frac{2(1-\gamma)^2}{3\mu_1 M(1-\gamma^N)^2\sigma^2}$ and $\delta \in (0,1)$ we have that with probability at least $1-\delta$ the following holds:

$$
\begin{aligned}
f^* - f_1 &\leqslant L \sum_{k=1}^{N} \sum_{i=1}^{k-1} \gamma^{k-i} \|d_i\|^2 - \sum_{k=1}^{N} \sum_{i=1}^{k} \gamma^{k-i} \mu_i \alpha_i \|\nabla f_i\|^2 \\
&\quad - \sum_{k=1}^{N} \sum_{i=1}^{k} \gamma^{k-i} \mu_i \alpha_i \nabla f_i^T \xi_i + \sum_{k=1}^{N} \frac{L}{2} \|d_k\|^2 \\
&\leqslant \frac{L}{1-\gamma} \sum_{k=1}^{N} \|d_k\|^2 - \sum_{k=1}^{N} \mu_k \alpha_k \|\nabla f_k\|^2 \\
&\quad + \frac{3\lambda \mu_1 M (1-\gamma^N)^2 \sigma^2}{4(1-\gamma)^2} \sum_{k=1}^{N} \mu_k \alpha_k \|\nabla f_k\|^2 + \frac{1}{\lambda} \ln \frac{1}{\delta} + \sum_{k=1}^{N} \frac{L}{2} \|d_k\|^2 \\
&\leqslant \frac{L(3-\gamma)}{(1-\gamma)^3} \sum_{k=1}^{N} \mu_k^2 \alpha_k^2 \|g_k\|^2 - \frac{1}{2} \sum_{k=1}^{N} \mu_k \alpha_k \|\nabla f_k\|^2 \\
&\quad + \frac{3\mu_1 M (1-\gamma^N)^2 \sigma^2}{2(1-\gamma)^2} \ln \frac{1}{\delta}.
\end{aligned} \tag{4.14}
$$

Multiplying by 2 and rearranging the last inequality we prove the lemma. □

The last technical result gives a probability bound for the noise variance needed for the main theorem.

Lemma 3. *Let assumption (A4) holds and let $\{x_k\}$ be a sequence generated by Algorithm 1. Let $\delta \in (0,1)$. Then with probability at least $1-\delta$ the following holds:*

$$\max_{1\leqslant k\leqslant N} \|g_k - \nabla f_k\|^2 \leqslant \sigma^2 \ln \frac{Ne}{\delta}$$

Proof. Follows immediately from Markov's inequality by applying assumption (A4). For detailed proof see the proof of Lemma 5 in [10]. □

Finally, we formulate the main statement.

Theorem 1. *Let assumptions (A1)–(A4) hold, $\mu_k = \frac{c}{\sqrt{k}}$, where $c \leqslant \frac{(1-\gamma)^3}{8LM(3-\gamma)}$ and let $\{x_k\}$ be a sequence generated by Algorithm 1. Let $\delta \in (0,1)$. Then in the case of an additive gradient noise (4.2), with probability at least $1-\delta$, the following holds:*

$$\min_{1\leqslant k\leqslant N} \|\nabla f_k\|^2 \leqslant \frac{1}{N}\sum_{k=1}^{N} \|\nabla f_k\|^2 \leqslant D_1 \frac{1}{\sqrt{N}} + D_2 \frac{\ln \frac{Ne}{\delta} \ln (Ne)}{\sqrt{N}},$$

where $D_1 = \frac{4(f_1 - f^*)}{cm} + \frac{6M(1-\gamma^N)^2\sigma^2}{m(1-\gamma)^2} \cdot \ln \frac{1}{\delta}$ and $D_2 = \frac{8L(3-\gamma)\sigma^2 M^2 c}{m(1-\gamma)^3}$.

Proof. Since $\|a+b\|^2 = \|a\|^2 + 2a^T b + \|b\|^2 \leqslant \|a\|^2 + 2\|a\|\|b\| + \|b\|^2 \leqslant 2\|a\|^2 + 2\|b\|^2$, using (4.2) we get:

$$\begin{aligned}
\sum_{k=1}^{N} \mu_k^2 \alpha_k^2 \|g_k\|^2 &= \sum_{k=1}^{N} \mu_k^2 \alpha_k^2 \|\nabla f_k + \xi_k\|^2 \\
&\leqslant 2\sum_{k=1}^{N} \mu_k^2 \alpha_k^2 \|\nabla f_k\|^2 + 2\sum_{k=1}^{N} \mu_k^2 \alpha_k^2 \|\xi_k\|^2 \\
&\leqslant 2\sum_{k=1}^{N} \mu_k^2 \alpha_k^2 \|\nabla f_k\|^2 + 2 \max_{1\leqslant k\leqslant N} \|\xi_k\|^2 \sum_{k=1}^{N} \mu_k^2 \alpha_k^2.
\end{aligned}$$

Now, substututing the last inequality in Lemma 2, and using Lemma 3, (4.3), $\mu_k \leqslant \mu_1 = c$ and $\sum_{k=1}^{N} \mu_k^2 \leqslant c^2 N \leqslant c^2 \ln(Ne)$, we have that for $\delta \in (0,1)$, with probability at least $1-\delta$, the following inequality stands:

$$\begin{aligned}
\sum_{k=1}^{N} \mu_k \alpha_k \|\nabla f_k\|^2 &\leqslant \frac{2L(3-\gamma)}{(1-\gamma)^3} \sum_{k=1}^{N} \mu_k^2 \alpha_k^2 \|g_k\|^2 + 2(f_1 - f^*) + \frac{3cM(1-\gamma^N)^2\sigma^2}{(1-\gamma)^2} \cdot \ln \frac{1}{\delta} \\
&\leqslant \frac{4L(3-\gamma)}{(1-\gamma)^3} \sum_{k=1}^{N} \mu_k^2 \alpha_k^2 \|\nabla f_k\|^2 + \frac{4L(3-\gamma)}{(1-\gamma)^3} \max_{1\leqslant k\leqslant N} \|\xi_k\|^2 \sum_{k=1}^{N} \mu_k^2 \alpha_k^2 \\
&\quad + 2(f_1 - f^*) + \frac{3cM(1-\gamma^N)^2\sigma^2}{(1-\gamma)^2} \cdot \ln \frac{1}{\delta} \\
&\leqslant \frac{4L(3-\gamma)cM}{(1-\gamma)^3} \sum_{k=1}^{N} \mu_k \alpha_k \|\nabla f_k\|^2 + \frac{4L(3-\gamma)}{(1-\gamma)^3} \sigma^2 \ln \frac{Ne}{\delta} M^2 c^2 \ln(Ne) \\
&\quad + 2(f_1 - f^*) + \frac{3cM(1-\gamma^N)^2\sigma^2}{(1-\gamma)^2} \cdot \ln \frac{1}{\delta}.
\end{aligned}$$

By rearranging the last inequality, we have that with probability at least $1-\delta$,

$$(1-\frac{4L(3-\gamma)cM}{(1-\gamma)^3})\sum_{k=1}^{N}\mu_k\alpha_k\|\nabla f_k\|^2 \leqslant \frac{4L(3-\gamma)\sigma^2M^2c^2}{(1-\gamma)^3}\ln\frac{Ne}{\delta}\ln(Ne) + 2(f_1-f^*)+\frac{3cM(1-\gamma^N)^2\sigma^2}{(1-\gamma)^2}\cdot\ln\frac{1}{\delta}.$$

Because of $c \leqslant \frac{(1-\gamma)^3}{8LM(3-\gamma)}$, we have that $1-\frac{4L(3-\gamma)cM}{(1-\gamma)^3} \geq \frac{1}{2}$, and the last inequality, $\mu_k \geq \mu_N = \frac{c}{\sqrt{N}}$ and $\alpha_k \geq m$ imply that with probability at least $1-\delta$,

$$\frac{cm}{2\sqrt{N}}\sum_{k=1}^{N}\|\nabla f_k\|^2 \leqslant \frac{4L(3-\gamma)\sigma^2M^2c^2}{(1-\gamma)^3}\ln\frac{Ne}{\delta}\ln(Ne) + 2(f_1-f^*)+\frac{3cM(1-\gamma^N)^2\sigma^2}{(1-\gamma)^2}\cdot\ln\frac{1}{\delta}.$$

By rearranging the last inequality, we have that with probability at least $1-\delta$,

$$\frac{1}{N}\sum_{k=1}^{N}\|\nabla f_k\|^2 \leqslant \frac{8L(3-\gamma)\sigma^2M^2c}{m(1-\gamma)^3}\cdot\frac{\ln\frac{Ne}{\delta}\ln(Ne)}{\sqrt{N}} + \left(\frac{4(f_1-f^*)}{cm}+\frac{6M(1-\gamma^N)^2\sigma^2}{m(1-\gamma)^2}\cdot\ln\frac{1}{\delta}\right)\cdot\frac{1}{\sqrt{N}}.$$

On the other hand

$$\frac{1}{N}\sum_{k=1}^{N}\|\nabla f_k\|^2 \geq \frac{1}{N}\cdot N\cdot\min_{1\leqslant k\leqslant N}\|\nabla f_k\|^2 = \min_{1\leqslant k\leqslant N}\|\nabla f_k\|^2,$$

which completes the proof. □

The high probability bound achieved in Theorem 1 shows that the convergence rate of Algorithm 1 is $\mathcal{O}(1/\sqrt{N})$, in noisy and noise-free environment ($\sigma = 0$). The proposed method in Algorithm 1 with BB information is theoretically comparable to the "vanilla" stochastic gradient method with momentum from [10] without BB information, see Theorem 1 in [10], but, as we will see in the next section, it has more promising numerical implementation when scaling of functions is an issue.

5. Numerical results

We performed numerical tests on a set of problems of different type and dimension. The details for test problems are given in Table 1. Of the test problems, four are quadratic functions (problems 1–4), two are convex exponential (problems 5 and 6) and one is polynomial of order greater than two (problem 7).

We compared performances of two methods, the first method is "vanilla" stochastic gradient method with momentum from [10] without BB information (SGM) and the proposed method in this paper in Algorithm 1, stochastic gradient method with momentum and BB information (SGMBB). To test for scale invariance, each

No.	Name	n	Reference
1	`quad`	2	$f(x) = 0.5x_1^2 + 2x_2^2$
2	`bb_quad`	4	The original BB test from [1]
3	`QuadSpec`	10	From [4]
4	`QuadNR`	10	From [4]
5	`strconvex1`	10	From [18]
6	`strconvex2`	10	From [18]
7	`variably`	4	Problem 25 from [12]

TABLE 1. Test-problems.

function was tested at seven scales ω of the kind $\omega = 10^p$ with $p \in \{-3, -2, \ldots, 3\}$. The noise was simulated and sampled from a zero-mean Normal distribution and standard deviation (noise level) $\sigma \in \{0.1, 0.5, 1.0\}$. A noise-free case ($\sigma = 0$) is also considered. For the non-increasing sequence $\{\mu_k\}$ in the both methods, we set $\mu_k = 1/\sqrt{k}$ for $k = 1, 2, \ldots$. Additionally, for SGMBB method (Algorithm 1) we set the BB step bounds $m \leqslant \alpha_k \leqslant M$, with $m = 10^{-6}$ and $M = 10^6$, to ensure positive not too small and not too large BB steps, as it is theoretically supported.

As performance metrics we use the number of iterations required for the algorithm to stop. Both algorithms stop if either the maximum number of iterations $N = 5000$ is reached, or the stochastic gradient has sufficiently decreased, i.e. $\|g_k\| \leq \varepsilon \|g_1\|$, with $\varepsilon = 10^{-3}$. In order for the results to be statistically meaningful, 50 runs are performed for each combination of a function, scale and noise level. From these we count the *non-divergent* runs (where appropriate, this number is given in parenthesis) and calculate the average number of iterations required for the algorithm to stop. If none of the 50 runs are successful, we denote it by NC. In noise-free case, only one run per different setting is performed, since all 50 runs will be the same. The results are reported in Tables 2–5.

In noise-free environment, as evidenced in Table 2, the SGMBB method is truly scale invariant: it takes the same number of iterations to converge for any of the test-problems. On the contrary, for the SGM method, it is clear that this is not the case. Besides scale invariance, the SGMBB method is almost always convergent, which is not the case with the SGM method, that does not converge in 23 of 49 different settings (Table 2). When noise is present, the SGMBB method has stable performance while the number of iterations stays fairly stable across the simulated tests for most problems (Tables 3–5).

Both methods seem to have a somewhat worse performance for small values of ω compared to the "baseline" value $w = 1$. Generally, for the larger values of ω, the SGMBB method outperforms the SGM method: the performance of SGMBB seems to improve as ω increases (likely due to the fact that the influence of the noise is lower relative to the function and gradient values), while the SGM fails to converge.

prb.	Method	$\omega = 0.001$	$\omega = 0.01$	$\omega = 0.1$	$\omega = 1$	$\omega = 10$	$\omega = 100$	$\omega = 1000$
1	SGM	5000	105	57	61	2773	NC	NC
	SGMBB	74	74	74	74	74	74	74
2	SGM	5000	880	102	753	NC	NC	NC
	SGMBB	162	162	162	162	162	162	162
3	SGM	88	189	NC	NC	NC	NC	NC
	SGMBB	85	85	85	85	85	85	85
4	SGM	80	182	NC	NC	NC	NC	NC
	SGMBB	100	100	100	100	100	100	100
5	SGM	5000	1866	87	63	5000	325	NC
	SGMBB	54	54	54	54	54	54	54
6	SGM	5000	2705	99	87	5000	329	NC
	SGMBB	81	81	81	81	81	81	81
7	SGM	4	NC	NC	NC	NC	NC	NC
	SGMBB	8	8	8	8	8	8	NC

TABLE 2. Comparison of performance in a noise free environment ($\sigma = 0$).

prb.	Method	$\omega = 0.001$	$\omega = 0.01$	$\omega = 0.1$	$\omega = 1$	$\omega = 10$	$\omega = 100$	$\omega = 1000$
1	SGM	3890.1 (50)	176.2 (50)	61.2 (50)	61 (50)	2773 (50)	NC	NC
	SGMBB	3809.7 (50)	303 (50)	69.5 (50)	59.4 (50)	70.7 (50)	74 (50)	74 (50)
2	SGM	5000 (50)	5000 (50)	5000 (50)	4644.2 (50)	NC	NC	NC
	SGMBB	5000 (50)	5000 (50)	5000 (50)	4406.6 (50)	188.2 (50)	164.1 (50)	163.6 (50)
3	SGM	5000 (50)	650 (50)	NC	NC	NC	NC	NC
	SGMBB	5000 (50)	154.7 (50)	86.7 (50)	85 (50)	85 (50)	85 (50)	85 (50)
4	SGM	2180 (50)	182 (50)	NC	NC	NC	NC	NC
	SGMBB	3617.2 (50)	104.2 (50)	98.6 (50)	104.4 (50)	103.7 (50)	100 (50)	100 (50)
5	SGM	5000 (50)	5000 (50)	5000 (50)	5000 (50)	5000 (50)	329.5 (50)	NC
	SGMBB	338.5 (50)	1162.6 (17)	2389.1 (32)	5000 (50)	76.8 (50)	54 (50)	54 (50)
6	SGM	5000 (50)	5000 (50)	5000 (50)	5000 (50)	5000 (50)	328.3 (50)	NC
	SGMBB	424.2 (49)	1575 (15)	898.4 (17)	5000 (50)	90.9 (50)	82 (50)	81 (50)
7	SGM	25.7 (34)	NC	NC	NC	NC	NC	NC
	SGMBB	154.3 (40)	7.8 (50)	8 (50)	8 (50)	8 (50)	8 (50)	NC

TABLE 3. Comparison of performance at noise level $\sigma = 0.1$.

6. CONCLUSIONS

The results we obtained demonstrate that it is possible to integrate the BB scheme into the momentum methods for unconstrained optimization. This allows to include at least some degree of an approximate second-order information into the optimization algorithms at a low additional per iteration cost. The advantage of the BB scheme is two-fold: firstly, the step sizes which include the second-order information are given in a closed form; secondly, the shape of the step sizes makes them naturally scale invariant. The algorithm we propose, the stochastic gradient momentum method with BB information, retains this property to a great degree even in noisy environments. The numerical experiments

prb.	Method	$\omega = 0.001$	$\omega = 0.01$	$\omega = 0.1$	$\omega = 1$	$\omega = 10$	$\omega = 100$	$\omega = 1000$
1	SGM	4506.4 (50)	2531.2 (50)	90.9 (50)	66.9 (50)	2773 (50)	NC	NC
	SGMBB	4085.8 (50)	3107.9 (50)	115 (50)	63.6 (50)	54.5 (50)	73.4 (50)	74 (50)
2	SGM	5000 (50)	5000 (50)	5000 (50)	5000 (50)	NC	NC	NC
	SGMBB	5000 (50)	5000 (50)	5000 (50)	5000 (50)	1517.6 (50)	170.2 (50)	161.8 (50)
3	SGM	5000 (50)	5000 (50)	NC	NC	NC	NC	NC
	SGMBB	5000 (50)	5000 (50)	91.8 (50)	85 (50)	85 (50)	85 (50)	85 (50)
4	SGM	5000 (50)	254.5 (50)	NC	NC	NC	NC	NC
	SGMBB	5000 (50)	240.3 (50)	101.8 (50)	101.1 (50)	105.5 (50)	102.2 (50)	100 (50)
5	SGM	5000 (50)	5000 (50)	5000 (50)	5000 (50)	5000 (50)	329 (50)	NC
	SGMBB	332.1 (19)	450.3 (38)	331.8 (17)	4907 (50)	5000 (50)	56.3 (50)	54 (50)
6	SGM	5000 (50)	5000 (50)	5000 (50)	5000 (50)	5000 (50)	328.4 (50)	NC
	SGMBB	327.9 (10)	369.2 (41)	1313.9 (19)	2767.3 (27)	4929.6 (50)	84.9 (50)	81.5 (50)
7	SGM	3321.6 (23)	NC	NC	NC	NC	NC	NC
	SGMBB	NC	16.4 (50)	8 (50)	8 (50)	8 (50)	8 (50)	NC

TABLE 4. Comparison of performance at noise level $\sigma = 0.5$.

prb.	Method	$\omega = 0.001$	$\omega = 0.01$	$\omega = 0.1$	$\omega = 1$	$\omega = 10$	$\omega = 100$	$\omega = 1000$
1	SGM	4350.9 (50)	3632 (50)	209.6 (50)	72.2 (50)	2773 (50)	NC	NC
	SGMBB	3995.9 (50)	3777.8 (50)	308 (50)	69.5 (50)	59.4 (50)	70.7 (50)	74 (50)
2	SGM	5000 (50)	5000 (50)	5000 (50)	5000 (50)	NC	NC	NC
	SGMBB	5000 (50)	5000 (50)	5000 (50)	5000 (50)	4427.1 (50)	187.7 (50)	164.1 (50)
3	SGM	5000 (50)	5000 (50)	NC	NC	NC	NC	NC
	SGMBB	5000 (50)	5000 (50)	154.7 (50)	86.7 (50)	85 (50)	85 (50)	85 (50)
4	SGM	5000 (50)	2904.8 (50)	NC	NC	NC	NC	NC
	SGMBB	5000 (50)	3440.1 (50)	104.2 (50)	98.6 (50)	104.4 (50)	103.7 (50)	100 (50)
5	SGM	5000 (50)	5000 (50)	5000 (50)	5000 (50)	5000 (50)	328.4 (50)	NC
	SGMBB	333 (1)	335.4 (50)	1002.6 (14)	2765.4 (27)	5000 (50)	76.8 (50)	54 (50)
6	SGM	5000 (50)	5000 (50)	5000 (50)	5000 (50)	5000 (50)	328.9 (50)	NC
	SGMBB	NC	326.7 (48)	1178 (11)	1273.5 (15)	5000 (50)	90.9 (50)	82 (50)
7	SGM	4436.7 (6)	1 (2)	NC	NC	NC	NC	NC
	SGMBB	NC	110.3 (38)	7.8 (50)	8 (50)	8 (50)	8 (50)	NC

TABLE 5. Comparison of performance at noise level $\sigma = 1$.

on the chosen set of problems lead us to conclude that the new method clearly outperforms the "vanilla" stochastic gradient method with momentum in most cases. The high probability convergence bounds for the proposed method are achieved under standard set of assumptions.

Acknowledgements. For the second author, this research is supported by Ss. Cyril and Methodius University of Skopje, North Macedonia scientific research project NIP.UKIM.20-21.6.

REFERENCES

[1] J. Barzilai, J. M. Borwein, *Two-Point Step Size Gradient Methods*, IMA Journal of Numerical Analysis. 8 (1988), 141–148.

[2] O. Burdakov, Y. H. Dai, N. Huang, *Stabilized Barzilai-Borwein method*, arXiv preprint https://arxiv.org/abs/1907.06409v3, 2019.
[3] A. Cutkosky, F. Orabona, *Momentum-Based Variance Reduction in Non-Convex SGD*, Proceedings of the 33rd International Conference on Neural Information Processing Systems, December 2019, Article No. 1365 (2019), 15236–15245.
[4] Y. H. Dai, Y. Huang, X. W. Liu, *A family of spectral gradient methods for optimization*, Computational Optimization and Applications, 74 (2019), 43–65.
[5] D. Di Serafino, G. Toraldo, M. Viola, *Using gradient directions to get global convergence of Newton-type methods*, Applied Mathematics and Computation, 409, 2021.
[6] J. Duchi, E. Hazan, Y. Singer, *Adaptive Subgradient Methods for Online Learning and Stochastic Optimization*, Journal of Machine Learning Research, 12 (2011), 2121–2159.
[7] M. Kempka, W. Kotlowski, M. K. Warmuth, *Adaptive scale-invariant online algorithms for learning linear models*, arXiv preprint https://arxiv.org/abs/1902.07528v1, 2019.
[8] D. P. Kingma, J. Ba, *Adam: A Method for Stochastic Optimization*, arXiv preprint https://arxiv.org/abs/1412.6980v9, 2014.
[9] N. Krejić, Z. Lužanin, I. Stojkovska, Z. Ovcin, *Descent direction method with line search for unconstrained optimization in noisy environment*, Optim Methods Softw 30(6) (2015), 1164-1184, DOI: 10.1080/10556788.2015.1025403.
[10] X. Li, F. Orabona, *A high probability analysis of adaptive SGD with momentum*, arXiv preprint https://arxiv.org/abs/2007.14294v1, 2020.
[11] Z. Liu, T. D. Nguyen, T. H. Nguyen, A. Ene, H. L. Nguyen, *High Probability Convergence of Stochastic Gradient Methods*, Proceedings of the 40th International Conference on Machine Learning, Honolulu, Hawaii, USA. PMLR 202, Article No.: 907, (2023), 21884–21914.
[12] J. J. More, B. S. Hilström, K. E. Garbow, *Testing Unconstrained Optimization Software*, ACM Transactions of Mathematical Software, 7(1) (1981), 17–41.
[13] Y. Nesterov, *A Method for Solving a Convex Programming Problem with Convergence Rate* $O(1/k^2)$, Soviet Mathematics Doklady, 27 (1983), 372–376.
[14] Y. Nesterov, *Introductory Lectures on Convex optimization: A Basic Course*, Springer Science+Business Media, New York, 2004.
[15] F. Orabona, *Simultaneous model selection and optimization through parameter-free stochastic learning*, Advances in Neural Information Processing Systeems, (27) (2014), 1116–1124.
[16] F. Orabona, D. Pal, *Scale-free algorithms for online linear optimization*, Algorithmic Learning Theory (2015), 287–301.
[17] B. T. Polyak, *Some methods of speeding up the convergence of iteration methods*, USSR Computational Mathematics and Mathematical Physics, 4 (5) (1964), 1–17.
[18] M. Raydan, *The Barzilai and Borwein Gradient Method for the Large Scale Unconstrained Minimization Problem*, SIAM J. Optim, 7(1) (1997), 26–33.
[19] H. Robbins, S. Monro, *A stochastic approximation method*, Ann. Math. Statist., 22 (1951), 400–407.
[20] S. Ruder, *An overview of gradient descent optimization algorithms*, arXiv preprint https://arxiv.org/abs/1609.04747, 2017.
[21] N. N. Schraudolph, J. Yu, S. Günter, A *Stochastic Quasi-Newton Method for Online Convex Optimization*, Proceedings of 11th International Conference on Artificial Intelligence and Statistics, San Juan, Puerto Rico (2007), 433–440.
[22] W. Sun, Y. X. Yuan, *Optimization Theory and Methods: Nonlinear Programming*, Springer, New York, 2006.
[23] C. Tan, Sh. Ma, Y-H. Dai, Y. Qian, *Barzilai-Borwein Step Size for Stochastic Gradient Descent*, Advances in Neural Information Processing Systems 29, 2016, https://proceedings.neurips.cc/paper_files/paper/2016/file/c86a7ee3d8ef0b551ed58e354a836-f2b-Paper.pdf.
[24] H. Tankaria, N. Yamashita, *A Stochastic Variance Reduced Gradient using Barzilai-Borwein Techniques as Second Order Information*, arXiv preprint https://arxiv.org/abs/2208.11075, 2022.
[25] X. Wang, S. Ma, D. Goldfarb, W. Liu, *Stochastic quasi-Newton methods for nonconvex stochastic optimization*, SIAM Journal on Optimization 27(2) (2017), 927–956.

[26] L. Wang, H. Wu, I. A. Matveev, *Stochastic Gradient Method with Barzilai-Borwein Step for Unconstrained Nonlinear Optimization*, Journal of Computer and Systems Sciences International, 60(1) (2021), 75–86.
[27] D. Zhou, J. Chen, Y. Cao, Y. Tang, Z. Yang, Q. Gu, *On the Convergence of Adaptive Gradient Methods for Nonconvex Optimization*, Transactions on Machine Learning Research (2024), 2835-8856.

FILIP NIKOLOVSKI
SS. CYRIL AND METHODIUS UNIVERSITY IN SKOPJE,
FACULTY OF MECHANICAL ENGINEERING,
RUGJER BOSHKOVIKJ 18, SKOPJE, NORTH MACEDONIA
Email address: filip.nikolovski@mf.edu.mk

IRENA STOJKOVSKA
SS. CYRIL AND METHODIUS UNIVERSITY IN SKOPJE,
FACULTY OF NATURAL SCIENCES AND MATHEMATICS,
ARHIMEDOVA 3, SKOPJE, NORTH MACEDONIA
Email address: irenatra@pmf.ukim.mk, irena.stojkovska@gmail.com

Received 30.10.2023
Revised 16.6.2024
Accepted 4.7.2024