BCOS: A Method for Stochastic Approximation

Anonymous Author(s)
Affiliation
Address
email

Abstract

We consider stochastic approximation with block-coordinate stepsizes and propose adaptive stepsize rules that aim to minimize the expected distance of the next iterate from an optimal point. These stepsize rules use online estimates of the second moment of the search direction along each block coordinate, and the popular Adam algorithm can be interpreted as using a particular heuristic for such estimation. By leveraging a simple conditional estimator, we derive variants of BCOS that obtain competitive performance but require fewer optimizer states and hyper-parameters. In addition, our convergence analysis relies on a simple aiming condition that assumes neither convexity nor smoothness, thus has broad applicability.

1 Introduction

We consider unconstrained stochastic optimization problems of the form

$$\underset{x\in\mathbf{R}^n}{\text{minimize}}\ F(x) := \mathbf{E}_\xi[f(x,\xi)], \tag{1}$$

where $x \in \mathbf{R}^n$ is the decision variable, ξ is a random variable, and f is the loss function. In the context of machine learning, x represents the parameters of a prediction model, ξ represents randomly sampled data, and $f(x,\xi)$ is the loss in making predictions about ξ using the parameters x.

Suppose that for any pair x and ξ, we can evaluate the gradient of f with respect to x, denoted as $\nabla f(x,\xi)$. Starting with an initial point $x_0 \in \mathbf{R}^n$, the classical *stochastic approximation* method [38] generates a sequence $\{x_1, x_2, \ldots\}$ with the update rule

$$x_{t+1} = x_t - \alpha_t \nabla f(x_t, \xi_t), \tag{2}$$

where α_t is the *stepsize*, which is often called the *learning rate* in the machine learning literature. The convergence properties of this method are well studied in the stochastic approximation literature [e.g., 38, 3, 6, 44, 52]. Despite the rich literature on their convergence theory, stochastic approximation methods in practice often require heuristics and trial and error in choosing the stepsize sequence $\{\alpha_t\}$. Adaptive rules that can adjust stepsizes on the fly have been developed in both the optimization literature [e.g., 10, 25, 33, 40, 41, 42, 43] and by the machine learning community [e.g., 22, 32, 46, 47]. More recently, adaptive algorithms that use *coordinate-wise* stepsizes have become very popular following the seminal works of AdaGrad [14] and Adam [26]. In this paper, we present a framework for better understanding such methods and propose a family of new, effective methods.

1.1 Stochastic approximation with block-coordinate stepsizes

We focus on stochastic approximation with *block-coordinate stepsizes*, specifically of the form

$$x_{t+1} = x_t - s_t \odot d_t, \tag{3}$$

where $d_t \in \mathbf{R}^n$ is a stochastic search direction, $s_t \in \mathbf{R}^n$ is a vector of coordinate-wise stepsizes, and $\odot$ denotes element-wise product (Hadamard product) of two vectors. The two most common

Submitted to 39th Conference on Neural Information Processing Systems (NeurIPS 2025). Do not distribute.

choices for the search direction are: the *stochastic gradient*, i.e., $d_t = \nabla f(x_t, \xi_t)$, and its *exponential moving average (EMA)*. Let $g_t = \nabla f(x_t, \xi_t)$, the EMA of stochastic gradient can be expressed as

$$d_t = \beta d_{t-1} + (1-\beta) g_t, \tag{4}$$

where $\beta \in [0, 1)$ is a smoothing factor. This is often called the *stochastic momentum*.

The Adam algorithm [26] uses the direction in (4) and sets the coordinate-wise stepsizes as

$$s_{t,i} = \alpha_t / (\sqrt{v_{t,i}} + \epsilon), \qquad i = 1, \dots, n, \tag{5}$$

where $\alpha_t \in \mathbf{R}$ is a common stepsize *schedule* and each $v_{t,i}$ is the EMA of the squared coordinate gradient $g_{t,i}^2$, with a *different*, often *larger*, smoothing factor $\beta' \in (0, 1)$. More specifically,

$$v_{t,i} = \beta' v_{t-1,i} + (1-\beta') g_{t,i}^2, \qquad i = 1, \dots, n. \tag{6}$$

Here $\epsilon > 0$ is a small constant to improve numerical stability when $v_{i,t}$ becomes very close to zero.

Adam [26] and its variant AdamW [31] have been very successful in training large-scale deep learning models. However, theoretical understanding of their convergence properties and empirical performance is still incomplete despite a lot of recent efforts [e.g., 37, 4, 1, 9, 56, 55, 28]. On the other hand, there have been many works that propose new variants or alternatives to Adam/AdamW, either starting from fundamental principles [e.g., 53, 17, 21, 29, 24] or based on empirical algorithm search [e.g., 5, 54] But all have limited success. Adam and especially AdamW are still the dominant algorithms for training large deep learning models, and their effectiveness remains a myth.

1.2 Contributions and outline

We propose a family of *block-coordinate optimistic stepsize* (BCOS) rules for stochastic approximation. BCOS provides a novel interpretation of Adam and AdamW and their convergence analysis as special cases of a general framework. Moreover, we derive variants of BCOS that obtain competitive performance but require fewer optimizer states and hyper-parameters. More specifically:

- In Section 2, we derive BCOS by minimizing the expected distance of the next iterate from an optimal point. While the optimal stepsizes cannot be computed exactly, we make optimistic simplifications and approximate the second moment of gradients with simple EMA estimators.
- In Section 3, we instantiate BCOS with specific search directions. In particular, we show that RMSprop [48] and Adam [26] can be interpreted as special cases of BCOS. By leveraging a simple conditional estimator, we derive new variants that require fewer optimizer states and hyper-parameters. Integrating with decoupled weight decay [31] gives the BCOSW variants.
- In Section 4, we present convergence analysis of BCOS(W) based on a simple aiming condition, which assumes neither convexity nor smoothness, thus has broad applicability. We obtain strong guarantees in terms of almost sure convergence, and characterize the effect of signal-to-noise ratio of the online estimators on the convergence behavior. Our results also apply to Adam(W).
- Finally, in Section 5, we present numerical experiments to compare BCOSW and AdamW on several Deep Learning tasks and demonstrate the effectiveness of the proposed methods.

1.3 Notations

Let $\mathcal{I}_1, \dots, \mathcal{I}_m$ be a non-overlapping partition of the coordinate index set $\{1, \dots, n\}$, each with cardinality $n_k = |\mathcal{I}_k|$. Correspondingly, we partition the vectors x_t, s_t and d_t into blocks $x_{t,k}$, $s_{t,k}$ and $d_{t,k}$ in $\mathbf{R}^{n_k}$ for $k = 1, \dots, m$. We use a common stepsize $\gamma_{t,k} \in \mathbf{R}$ within each block, i.e., $s_{t,k} = \gamma_{t,k} \mathbf{1}_{n_k}$. As a result, the explicit block-coordinate update form of (3) can be written as

$$x_{t+1,k} = x_{t,k} - s_{t,k} \odot d_{t,k} = x_{t,k} - \gamma_{t,k} d_{t,k}, \qquad k = 1, \dots, m.$$

Notice that $\gamma_{t,k}$ is always a scalar and γ_t is a vector in $\mathbf{R}^m$ instead of $\mathbf{R}^n$ (unless $m = n$).

Throughout this paper, $\langle \cdot, \cdot \rangle$ denotes the standard inner product in $\mathbf{R}^n$ and $\| \cdot \|$ the induced Euclidean norm. The signum function is defined as $\operatorname{sign}(\alpha) = 1$ if $\alpha > 0$, -1 if $\alpha < 0$ and 0 if $\alpha = 0$.

2 Derivation of BCOS

We first derive the ideal optimal stepsizes for block-coordinate update, which is not computable in practice; then we make several simplifications and approximations to derive the practical ones.

2.1 Block-coordinate optimal stepsizes

We consider the change of distance to an optimal point x_* after one iteration of the algorithm (3):

$$\begin{aligned}\|x_{t+1} - x_*\|^2 &= \|x_t - s_t \odot d_t - x_*\|^2 \\ &= \|x_t - x_*\|^2 - 2\langle x_t - x_*, s_t \odot d_t\rangle + \|s_t \odot d_t\|^2.\end{aligned}$$

Exploiting the block partitions of x_t, s_t and d_t and using $s_{t,k} = \gamma_{t,k}\mathbf{1}_{n_k}$, we obtain

$$\|x_{t+1} - x_*\|^2 = \|x_t - x_*\|^2 + \sum_{k=1}^{m} \left(-2\gamma_{t,k}\langle x_{t,k} - x_{*,k},\, d_{t,k}\rangle + \gamma_{t,k}^2\|d_{t,k}\|^2\right).$$

Taking expectation conditioned on the realization of all random variables up to x_t, i.e.,

$$\mathbf{E}_t[\cdot] := \mathbf{E}[\cdot|x_0, d_0, x_1, d_1, \ldots, x_t], \tag{7}$$

we have

$$\mathbf{E}_t\left[\|x_{t+1} - x_*\|^2\right] = \|x_t - x_*\|^2 + \sum_{k=1}^{m}\Big(-2\gamma_{t,k}\big\langle x_{t,k} - x_{*,k},\, \mathbf{E}_t[d_{t,k}]\big\rangle + \gamma_{t,k}^2\mathbf{E}_t\left[\|d_{t,k}\|^2\right]\Big). \tag{8}$$

In order to minimize the expected distance from x_{t+1} to x_*, we can minimize the right-hand side of (8) over the stepsizes $\{\gamma_{t,k}\}_{k=1}^m$. This results in the *optimal* stepsizes

$$\widehat{\gamma}_{t,k} = \frac{\langle x_{t,k} - x_{*,k},\, \mathbf{E}_t[d_{t,k}]\rangle}{\mathbf{E}_t[\|d_{t,k}\|^2]}, \qquad k = 1, \ldots, m. \tag{9}$$

Notice that these optimal stepsizes can be positive or negative, depending on the sign of the inner product in the numerator. Apparently, they are not computable in practice, because we do not have access of x_* and cannot evaluate the expectations precisely. We address this issue in the next section.

2.2 Block-coordinate optimistic stepsizes

We need to make several simplifications and approximations to derive a practical stepsize rule. Our first step aims to avoid the direct reliance on x_*. To this end, we rewrite the numerator in (9) as

$$\big\langle x_{t,k} - x_{*,k},\, \mathbf{E}_t[d_{t,k}]\big\rangle = \|x_{t,k} - x_{*,k}\|\|\mathbf{E}_t[d_{t,k}]\|\cos\theta_{t,k},$$

where $\theta_{t,k}$ is the angle between the two vectors $x_{t,k} - x_{*,k}$ and $\mathbf{E}_t[d_{t,k}]$. We absorb the quantities related to $x_{*,k}$ into a tunable parameter $\alpha_{t,k} \approx \|x_{t,k} - x_{*,k}\|\cos\theta_{t,k}$, which gives the stepsizes

$$\widetilde{\gamma}_{t,k} = \frac{\alpha_{t,k}\|\mathbf{E}_t[d_{t,k}]\|}{\mathbf{E}_t[\|d_{t,k}\|^2]}, \qquad k = 1, \ldots, m. \tag{10}$$

We emphasize that any $\alpha_{t,k}$ we choose in practice may only be a (very rough) approximation of $\|x_{t,k} - x_{*,k}\|\cos\theta_{t,k}$. In particular, while the optimal stepsizes $\widehat{\gamma}_{t,k}$ can be positive or negative, in practice it is very hard to estimate the sign of the inner product $\langle x_{t,k} - x_{*,k},\, \mathbf{E}_t[d_{t,k}]\rangle$. Instead, we take the pragmatic approach of restricting $\alpha_{t,k} > 0$, effectively being *optimistic* that the expected search directions $-\mathbf{E}_t[d_{t,k}]$ always point towards $x_{*,k}$ for all $k = 1, \ldots, m$.

A further simplification is to use a common stepsize schedule α_t across all blocks. This is often a reasonable choice for deep learning, where the model parameters are initialized randomly coordinate-wise such that $\mathbf{E}[\|x_{0,k}\|]$ is constant for each coordinate k [e.g., 13, 19]. This brings us to

$$\widetilde{\gamma}_{t,k} = \frac{\alpha_t\|\mathbf{E}_t[d_{t,k}]\|}{\mathbf{E}_t[\|d_{t,k}\|^2]}, \qquad k = 1, \ldots, m. \tag{11}$$

We note that with some abuse of notation, here α_t denotes a scalar, not a vector of $(\alpha_{t,1}, \ldots, \alpha_{t,k})$. This simplification reveals the connection between α_t and the distance $\|x_t - x_*\|$. Therefore, we expect α_t to decrease as $\|x_t - x_*\|$ gradually shrinks. A simple strategy is to use a monotonic stepsize schedule on α_t, such as the popular cosine decay [30] or linear decay [8].

Next, we need to replace the conditional expectations $\mathbf{E}_t[d_{t,k}]$ and $\mathbf{E}_t[\|d_{t,k}\|^2]$ in (11) with computable approximations. We adopt the conventional approach of exponential moving average (EMA):

$$\begin{aligned}u_{t,k} &= \beta u_{t-1,k} + (1-\beta)d_{t,k} \\ v_{t,k} &= \beta v_{t-1,k} + (1-\beta)\|d_{t,k}\|^2\end{aligned} \tag{12}$$

where $\beta \in [0, 1)$ is the smoothing factor. This leads to a set of practical stepsizes:

$$\gamma_{t,k} = \alpha_t\frac{\|u_{t,k}\|}{v_{t,k} + \epsilon}, \qquad k = 1, \ldots, m, \tag{13}$$

where we added a small constant $\epsilon > 0$ in the denominator to improve numerical stability.

Algorithm 1 BCOS-g

input: x_0, $\{\alpha_t\}_{t\geq 0}$, $\beta \in [0,1)$, $\epsilon > 0$
$v_{-1} = g_0^2$
for $t = 0, 1, 2, \ldots$ **do**
  $g_t = \nabla f(x_t, \xi_t)$
  $v_t = \beta v_{t-1} + (1-\beta) g_t^2$
  $x_{t+1} = x_t - \alpha_t \frac{g_t}{\sqrt{v_t}+\epsilon}$

(same as RMSprop [49])

Algorithm 2 BCOS-m

input: x_0, $\{\alpha_t\}$, $\beta_1, \beta_2 \in [0,1)$, $\epsilon > 0$
$m_{-1} = g_0,\ v_{-1} = g_0^2$
for $t = 0, 1, 2, \ldots$ **do**
  $g_t = \nabla f(x_t, \xi_t)$
  $m_t = \beta_1 m_{t-1} + (1-\beta_1) g_t$
  $v_t = \beta_2 v_{t-1} + (1-\beta_2) m_t^2$
  $x_{t+1} = x_t - \alpha_t \frac{m_t}{\sqrt{v_t}+\epsilon}$

2.3 Further simplification with one EMA estimator

The BCOS stepsizes in (13) are computed through the ratio of two online estimators $\|u_{t,k}\|$ and $v_{t,k}$, which are susceptible to large variations because the numerator and denominator may fluctuate in different directions. In this section, we derive a simplified stepsize rule that depends only on $v_{t,k}$.

First, recall the mean-variance decomposition of the conditional second moment,

$$\mathbf{E}_t[\|d_{t,k}\|^2] = \|\mathbf{E}_t[d_{t,k}]\|^2 + \mathbf{E}_t[\|d_{t,k} - \mathbf{E}_t[d_{t,k}]\|^2] = \|\mathbf{E}_t[d_{t,k}]\|^2 + \mathrm{Var}_t(d_{t,k}).$$

We interpret $\|\mathbf{E}_t[d_{t,k}]\|^2$ as the signal power and $\mathrm{Var}_t(d_{t,k})$ as the noise power, and define the *signal fraction* (SiF) as

$$\rho_{t,k} = \frac{\|\mathbf{E}_t[d_{t,k}]\|^2}{\mathbf{E}_t[\|d_{t,k}\|^2]} = \frac{\|\mathbf{E}_t[d_{t,k}]\|^2}{\|\mathbf{E}_t[d_{t,k}]\|^2 + \mathrm{Var}_t(d_{t,k})}. \tag{14}$$

Apparently we have $\rho_{t,k} \in [0,1]$. Using SiF, we can decompose the stepsizes in (10) as

$$\widetilde{\gamma}_{t,k} = \alpha_{t,k} \frac{\|\mathbf{E}_t[d_{t,k}]\|}{\mathbf{E}_t[\|d_{t,k}\|^2]} = \alpha_{t,k} \sqrt{\frac{\|\mathbf{E}_t[d_{t,k}]\|^2}{\mathbf{E}_t[\|d_{t,k}\|^2]}} \frac{1}{\sqrt{\mathbf{E}_t[\|d_{t,k}\|^2]}} = \frac{\alpha_{t,k}\sqrt{\rho_{t,k}}}{\sqrt{\mathbf{E}_t[\|d_{t,k}\|^2]}}. \tag{15}$$

Now we can merge $\sqrt{\rho_{t,k}} \in [0,1]$ into the tunable parameters $\alpha_{t,k}$ and let $\alpha'_{t,k} := \alpha_{t,k}\sqrt{\rho_{t,k}}$. Then, following the same arguments as in Section 2.2, we arrive at the following simplified stepsize rule:

$$\gamma_{t,k} = \alpha'_t \frac{1}{\sqrt{v_{t,k}} + \epsilon}, \qquad k = 1, \ldots, m, \tag{16}$$

where α'_t is a *scalar* stepsize schedule, and $v_{t,k}$ is given in (12). The similarity between Adam and BCOS in (16) is apparent, and we will explain their connection in detail in the next section.

3 Instantiations of BCOS

The derivation of BCOS in Section 2 is carried out with a general search direction d_t. In this section, we instantiate BCOS with two common choices of the search direction: the stochastic gradient and its EMA, also known as the *stochastic momentum*.

To simplify presentation, we focus on the case of *single coordinate blocks*, i.e., $m = n$ and $\mathcal{I}_k = \{k\}$ for $k = 1, \ldots, n$. Then we can express the EMA estimators for $\mathbf{E}_t[d_{t,k}^2]$ in a vector form:

$$v_t = \beta v_{t-1} + (1-\beta) d_t^2, \tag{17}$$

where d_t^2 denotes the element-wise squared vector $d_t \odot d_t$. We also have $s_t = \gamma_t \in \mathbf{R}^n$ and therefore

$$x_{t+1} = x_t - \gamma_t \odot d_t,$$

where the vector of coordinate-wise stepsizes, γ_t, can be expressed as

$$\gamma_t = \alpha_t \frac{1}{\sqrt{v_t} + \epsilon}. \tag{18}$$

Here $\sqrt{v_t}$ denotes element-wise square roots, $\sqrt{v_t} + \epsilon$ means element-wise addition of ϵ, and the fraction represent element-wise division or reciprocal. Again, the stepsize schedule α_t is a scalar. We no longer distinguish between α_t and α'_t because they are both tunable hyper-parameters.

Algorithm 3 BCOS-c

input: x_0, $\{\alpha_t\}_{t\geq 0}$, $\beta \in [0,1)$, $\epsilon > 0$
$m_{-1} = g_0,\ v_{-1} = g_0^2$
for $t = 0, 1, 2, \ldots$ **do**
  $g_t = \nabla f(x_t, \xi_t)$
  $m_t = \beta m_{t-1} + (1-\beta) g_t$
  $v_t = \big(1-(1-\beta)^2\big) m_{t-1}^2 + (1-\beta)^2 g_t^2$
  $x_{t+1} = x_t - \alpha_t \frac{m_t}{\sqrt{v_t}+\epsilon}$

Algorithm 4 BCOSW-c

input: x_0, $\{\alpha_t\}_{t\geq 0}$, $\beta \in [0,1)$, $\epsilon > 0$
$m_{-1} = g_0,\ v_{-1} = g_0^2$
for $t = 0, 1, 2, \ldots$ **do**
  $g_t = \nabla f(x_t, \xi_t)$
  $m_t = \beta m_{t-1} + (1-\beta) g_t$
  $v_t = \big(1-(1-\beta)^2\big) m_{t-1}^2 + (1-\beta)^2 g_t^2$
  $x_{t+1} = (1-\alpha_t\lambda) x_t - \alpha_t \frac{m_t}{\sqrt{v_t}+\epsilon}$

3.1 BCOS with EMA estimators

BCOS-g Algorithm 1 is the instantiation of BCOS using $\nabla f(x_t, \xi_t)$ as the search direction. We call it BCOS-g to signify the use of gradient as search direction. The vector v_t consists of coordinate-wise EMA estimators for $\mathbf{E}[g_{t,k}^2]$, and the notation $\frac{m_t}{\sqrt{v_t}+\epsilon}$ means element-wise division.

We immediately recognize that BCOS-g is exactly the RMSprop algorithm [49], which is one of the first effective algorithms to train deep learning models. Our BCOS framework gives a novel interpretation of RMSprop and its effectiveness. In the special case with $\beta = 0$ and $\epsilon = 0$, we have $v_t = g_t^2$, and both BCOS-g becomes the sign gradient method $x_{t+1} = x_t - \alpha_t \operatorname{sign}(g_t)$, which also received significant attention in the literature [35, 2, 45, 23].

BCOS-m Using the stochastic momentum as search direction has a long history in stochastic approximation [e.g., 18, 34, 40]. It has become the default option for modern deep learning due to its superior performance compared with using plain stochastic gradients. Following the standard notation in machine learning, we use m_t to denote the momentum, as shown in Algorithm 2. We call it BCOS-m to signify the use of momentum as the search direction. BCOS-m employs a second smoothing factor β_2 to calculate the EMA of m_t^2. These two smoothing factors β_1 and β_2 do not need to be the same and can be chosen independently in practice.

We notice that BCOS-m is very similar to Adam as given in (5) and (6). The difference is that in Adam, v_t is the EMA of g_t^2 instead of m_t^2. From BCOS perspective, Adam has a mismatch between the search direction m_t and the second moment estimator based on g_t^2, which must be compensated for by a larger smoothing factor β_2 (because m_t itself is a smoothed version of g_t). For BCOS-m, using $\beta_2 = \beta_1$ produces as good performance as Adam with the best tuned β_2 (see Section 5).

3.2 BCOS with conditional estimators

Recall that the optimal stepsizes $\widehat{\gamma}_{t,k}$ in (9) and their simplifications $\widetilde{\gamma}_{t,k}$ in (11) and (15) are all based on *conditional* expectation. In Section 3.1, we used coordinate-wise EMA of d_t^2 to approximate the conditional expectation $\mathbf{E}_t[d_t^2]$, i.e., v_t as estimator of $\mathbf{E}_t[d_t^2]$ in BCOS-g and of $\mathbf{E}_t[m_t^2]$ in BCOS-m, respectively. In this section, we show that with m_t as the search direction, we can exploit its update form to derive effective *conditional estimators* that can avoid using EMA.

We first repeat the definition of momentum here: $m_t = \beta m_{t-1} + (1-\beta) g_t$ with $\beta \in [0,1)$. To derive an estimator of $\mathbf{E}_t[m_t^2]$, we expand the square and take expectation of each term:

$$\begin{aligned}
\mathbf{E}_t\big[m_t^2\big] &= \mathbf{E}_t\big[(\beta m_{t-1} + (1-\beta) g_t)^2\big] \\
&= \beta^2 \mathbf{E}_t[m_{t-1}^2] + 2\beta(1-\beta)\mathbf{E}_t[m_{t-1} \odot g_t] + (1-\beta)^2 \mathbf{E}_t[g_t^2] \\
&= \beta^2 m_{t-1}^2 + 2\beta(1-\beta) m_{t-1} \odot \mathbf{E}_t[g_t] + (1-\beta)^2 \mathbf{E}_t[g_t^2], \qquad (19)
\end{aligned}$$

where we used the fact $\mathbf{E}_t[m_{t-1}^2] = m_{t-1}^2$ and $\mathbf{E}_t[m_{t-1}] = m_{t-1}$ thanks to the definition of $\mathbf{E}_t[\cdot]$ in (7). It remains to approximate $\mathbf{E}_t[g_t]$ and $\mathbf{E}_t[g_t^2]$. Clearly a good estimator for $\mathbf{E}_t[g_t]$ is m_t. To approximate $\mathbf{E}_t[g_t^2]$, we could use a separate EMA estimator $v_t' = \beta' v_{t-1}' + (1-\beta') g_t^2$, but this introduces another algorithmic state v_t' and a second smoothing factor β'. Meanwhile, we notice that the factor $(1-\beta)^2$ multiplying $\mathbf{E}_t[g_t^2]$ is usually very small, especially for β close to 1. As a result, any error in approximating $\mathbf{E}_t[g_t^2]$ is attenuated by a very small factor, so it may not cause much

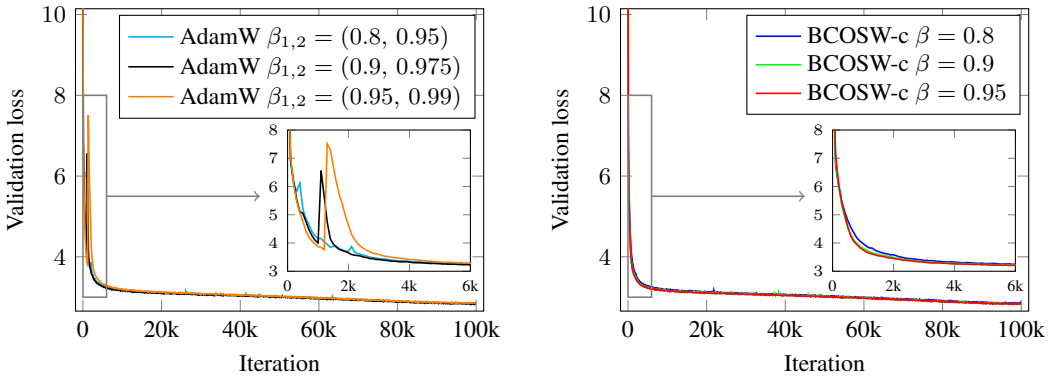

Figure 1: Comparing AdamW and BCOSW-c with different momentum parameters.

difference. Therefore, for simplicity, we choose to approximate $\mathbf{E}_t[g_t^2]$ with g_t^2 itself. Combining with approximating $\mathbf{E}_t[g_t]$ with m_t, we arrive at the following *conditional* estimator for $\mathbf{E}_t[m_t^2]$:

$$v_t = \beta^2 m_{t-1}^2 + 2\beta(1-\beta) m_{t-1} \odot m_t + (1-\beta)^2 g_t^2. \tag{20}$$

While this can be a very effective estimator, we derive another one that is much simpler and as effective. The key is to approximate $\mathbf{E}[g_t]$ in (19) with m_{t-1} instead of m_t, which results in

$$\begin{aligned} v_t &= \beta^2 m_{t-1}^2 + 2\beta(1-\beta) m_{t-1}^2 + (1-\beta)^2 g_t^2 \\ &= \left(1 - (1-\beta)^2\right) m_{t-1}^2 + (1-\beta)^2 g_t^2. \end{aligned} \tag{21}$$

It *resembles* the standard EMA estimator in Adam, shown in (6), with an effective smoothing factor

$$\beta' = 1 - (1-\beta)^2,$$

but with v_{t-1} *replaced by* m_{t-1}^2. As a result, the estimator in (21) does not need to store v_{t-1}, thus requiring fewer optimizer states. This also explains that the second smoothing factor in Adam, β_2, corresponding to β' here, should be much larger or closer to 1 than β. Specifically, $\beta = 0.9$ roughly corresponds to $\beta' = 0.99$. The estimator in (21) eliminates β_2 as a second hyper-parameter.

Finally, replacing v_t in BCOS-m with the one in (21) produces Algorithm 3. We call it BCOS-c to signify the *conditional* estimator. It has fewer optimizer states and fewer hyper-parameters to tune.

3.3 BCOS with decoupled weight decay

Weight decay is a common practice in training deep learning models to obtain better generalization performance. It can be understood as adding an L_2 regularization to the loss function, i.e., minimizing the regularized loss $\mathbf{E}_\xi[f(x,\xi)] + \frac{\lambda}{2}\|x\|^2$. Effectively, the stochastic gradient at x_t becomes $\nabla f(x_t, \xi_t) + \lambda x_t$. We can apply the BCOS family of algorithms by simply replacing $g_t = \nabla f(x_t, \xi_t)$ with $g_t = \nabla f(x_t, \xi_t) + \lambda x_t$. But a more effective way is to use *decoupled weight decay* as proposed in the AdamW algorithm [31]. Specifically, we apply weight decay separately in the BCOS update:

$$x_{t+1} = x_t - \gamma \odot d_t - \alpha_t \lambda x_t = (1 - \alpha_t \lambda) x_t - \gamma \odot d_t.$$

We call the resulting method BCOSW following the naming convention of AdamW. Algorithm 4 shows BCOSW with the conditional estimator. Other variants (-g and -m) can be obtained similarly. A PyTorch implementation of all BCOS and BCOSW variants is given in Appendix A.

4 Convergence analysis

In this section, we present the convergence analysis of BCOS and BCOSW. Due to space limit, we focus on BCOSW and give comments on BCOS wherever apply. Our analysis consists of two stages. First, we analyze the convergence properties of the *conceptual* BCOSW method

$$x_{t+1} = (1 - \alpha_t \lambda) x_t - \widetilde{\gamma}_t \odot d_t, \qquad \text{where} \qquad \widetilde{\gamma}_t = \alpha_t \frac{1}{\sqrt{\mathbf{E}_t[d_t^2]}}. \tag{22}$$

It is called "conceptual" because we cannot compute $\mathbf{E}_t[d_t^2]$ exactly in practice. Then, for the practical BCOSW algorithm with stepsize γ_t in (18), we bound the difference between the expected steps $\mathbf{E}_t[\gamma_t \odot d_t]$ and $\mathbf{E}_t[\widetilde{\gamma}_t \odot d_t] = \widetilde{\gamma}_t \odot \mathbf{E}_t[d_t]$, which produces the desired convergence guarantee.

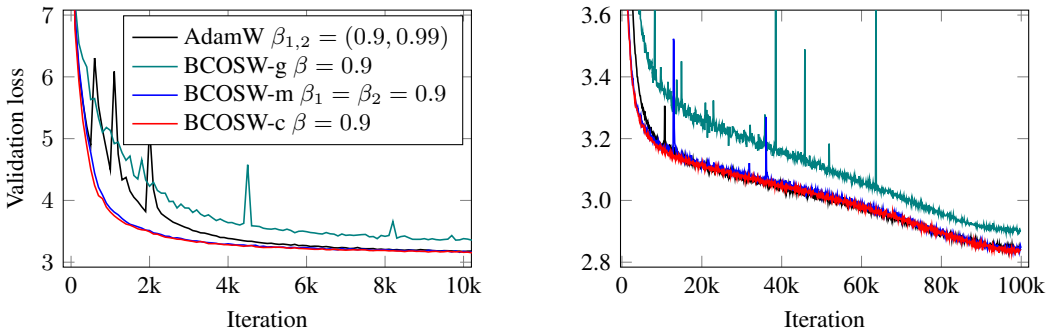

Figure 2: Comparing AdamW and BCOSW. Left: first 10k iterations; Right: all 100k iterations.

First, we need an appropriate condition to build our analysis. For the algorithm $x_{t+1} = x_t - \widetilde{\gamma}_t \odot d_t$, the next iterate x_{t+1} moves closer to x_* in expectation if the expected direction $-\mathbf{E}_t[\widetilde{\gamma}_t \odot d_t]$ aims towards x_* and α_t (a scalar) is sufficiently small. For the conceptual BCOS method, we have

$$\mathbf{E}_t[\widetilde{\gamma}_t \odot d_t] = \mathbf{E}_t\left[\alpha_t \frac{d_t}{\sqrt{\mathbf{E}_t[d_t^2]}}\right] = \alpha_t \frac{\mathbf{E}_t[d_t]}{\sqrt{\mathbf{E}_t[d_t^2]}} = \alpha_t \sqrt{\frac{\mathbf{E}_t[d_t]^2}{\mathbf{E}_t[d_t^2]}} \operatorname{sign}(\mathbf{E}_t[d_t]),$$

where $\operatorname{sign}(\cdot)$ denotes element-wise sign function. Recall the definition of SiF in (14). With single coordinate blocks, we can write the vector of coordinate-wise SiFs as $\rho_t = \frac{\mathbf{E}_t[d_t]^2}{\mathbf{E}_t[d_t^2]} \in [0,1]^n$. Then we have the expected update direction $\mathbf{E}_t[\widetilde{\gamma}_t \odot d_t] = \alpha_t \sqrt{\rho_t} \odot \operatorname{sign}(\mathbf{E}_t[d_t])$. Since $\alpha_t > 0$ is a scalar, we omit it from the statement of the aiming condition below.

Assumption A (Aiming condition)**.** *There exists $x_* \in \mathbf{R}^n$ such that*

$$\left\langle x_t - x_*,\ \sqrt{\rho_t} \odot \operatorname{sign}(\mathbf{E}_t[d_t]) + \lambda x_t \right\rangle \geq \lambda \|x_t - x_*\|^2 \tag{23}$$

holds for all $t \geq 0$ almost surely. If d_t is independent of the past trajectory conditioned on x_t, i.e., $\mathbf{E}_t[d_t] = \mathbf{E}[d_t|x_t]$, then it suffices to have (23) *hold for every $x \in \mathbf{R}^n$ (independent of the trajectory).*

Notice that we have $\mathbf{E}_t[d_t] = \mathbf{E}[d_t|x_t]$ when, e.g., $d_t = \nabla f(x_t, \xi_t)$ and ξ_t is independent of x_t. The aiming conditions assume neither convexity nor smoothness, but it has some overlapping characteristics with convexity, which we discuss in Appendix B.

4.1 Analysis of conceptual BCOSW

Our first result concerns the one-step contraction property of the conceptual algorithm in (22).

Lemma 4.1. *Suppose Assumption A holds, $\alpha_t \geq 0$ and $\alpha_t \lambda < 1$. Then we have*

$$\mathbf{E}_t\left[\|x_{t+1} - x_*\|^2\right] \leq (1 - \alpha_t \lambda)^2 \|x_t - x_*\|^2 + \alpha_t^2 c_*, \tag{24}$$

where $c_* = n + \lambda^2 \|x_*\|^2 + 2\lambda \|x_*\|_1$. Thus for sufficiently small α_t, $\mathbf{E}_t\left[\|x_{t+1} - x_*\|^2\right] \leq \|x_t - x_*\|^2$.

In fact, we can prove the following much stronger result of almost sure (a.s.) convergence.

Theorem 4.1. *Suppose the stepsize schedule $\{\alpha_t\}_{t\geq 0}$ and weight decay parameter λ satisfy*

$$\alpha_t \geq 0, \quad 0 \leq \alpha_t \lambda \leq 1, \quad \forall\, t \geq 0, \qquad \text{and} \qquad \textstyle\sum_{t=0}^{\infty} \alpha_t = \infty, \qquad \sum_{t=0}^{\infty} \alpha_t^2 < \infty. \tag{25}$$

Then Assumption A implies $\|x_t - x_*\| \to 0$ a.s. for the conceptual BCOSW method (22)*.*

In terms of convergence rate, we can readily obtain linear convergence to a neighborhood of x_* with a constant α_t based on (24). In addition, we have the following result on sublinear convergence.

Theorem 4.2. *Consider the conceptual BCOSW method* (22) *with the stepsize schedule $\alpha_t = \frac{\alpha}{t+1}$ where $1/2 < \alpha\lambda < 1$ is satisfied. Then Assumption A implies that for all $t \geq 1$,*

$$\mathbf{E}[\|x_t - x_*\|^2] \leq \frac{\alpha^2\left(c_* + \lambda^2 \mathbf{E}[\|x_0 - x_*\|^2] + \pi^2 \alpha^2 \lambda^2 c_*/6\right)}{2\alpha\lambda - 1} \frac{1}{t} + \mathcal{O}\left(\frac{1}{t^2} + \frac{1}{t^{2\alpha\lambda}}\right).$$

Without decoupled weight decay, BCOS may also have almost-sure convergence if the aiming condition with $\lambda = 0$ holds with strict inequality for $x_t \neq x_*$. However, the $\mathcal{O}(1/t)$ convergence rate no longer holds. The proofs of the above results are given in Appendix C.

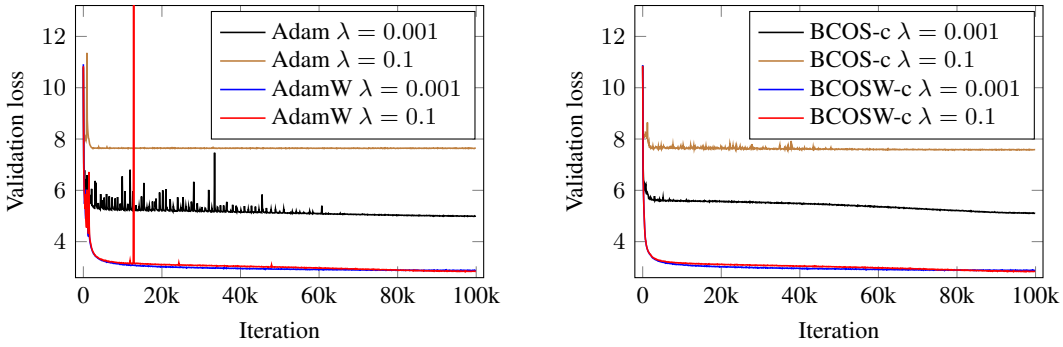

Figure 3: Left: Adam/ AdamW with $\beta_{1,2} = (0.9,\, 0.99)$. Right: BCOS/BCOSW with $\beta = 0.9$.

4.2 Analysis of practical BCOSW

Now we consider the practical BCOSW method $x_{t+1} = (1 - \alpha_t\lambda)x_t - \gamma_t \odot d_t$ with the stepsize vector γ_t given in (18). Our analysis is based on bounding the difference between the expected practical update $\mathbf{E}_t[\gamma_t \odot d_t]$ and the expected conceptual update $\mathbf{E}_t[\widetilde{\gamma}_t \odot d_t]$. Intuitively, it boils down to the quality of the estimator v_t. Specifically, we need the following assumption on its bias.

Assumption B. *There exists $\tau > 0$ and $\epsilon > 0$ such that for all $t \geq 0$ it holds that*

$$\left|\mathbf{E}_t[v_t] - \mathbf{E}_t[d_t^2]\right| \leq \tau \mathbf{E}_t[d_t^2] + \epsilon. \tag{26}$$

Based on this assumption, we have the following bound on the expected update directions.

Lemma 4.2. *Under Assumptions B, we have the following bound at each iteration t:*

$$\left|\frac{\mathbf{E}_t[d_t]}{\sqrt{\mathbf{E}_t[d_t^2]}} - \mathbf{E}_t\left[\frac{d_t}{\sqrt{v_t + \epsilon}}\right]\right| \leq c_t \left|\frac{\mathbf{E}_t[d_t]}{\sqrt{\mathbf{E}_t[d_t^2]}}\right| + \mathcal{O}(\epsilon) + \mathcal{O}(\mathrm{Var}_t(v_t)), \tag{27}$$

where $\mathcal{O}(\mathrm{Var}_t(v_t))$ includes terms such as $\mathbf{E}_t[(d_t - \mathbf{E}_t[d_t])(v_t - \mathbf{E}_t[v_t])^2]$ and $\mathbf{E}_t[(v_t - \mathbf{E}_t[v_t])^3]$ and higher-order terms. The coefficient c_t is defined as

$$c_t := \frac{4\tau + 3\tau^2}{8} + \frac{8 + 4\tau + 3\tau^2}{16}\left(\frac{1}{\mathrm{SNR}_t(v_t + \epsilon)} + \frac{1}{\sqrt{\mathrm{SNR}_t(d_t)}\sqrt{\mathrm{SNR}_t(v_t + \epsilon)}}\right). \tag{28}$$

Here, $\mathrm{SNR}_t(\cdot)$ denotes *conditional Signal-to-Noise Ratio*. Specifically, $\mathrm{SNR}_t(d_t) = \frac{\mathbf{E}_t[d_t]^2}{\mathrm{Var}_t(d_t)} = \frac{\rho_t}{1-\rho_t}$ and $\mathrm{SNR}_t(v_t + \epsilon) = \frac{\mathbf{E}[v_t+\epsilon]^2}{\mathrm{Var}_t(v_t+\epsilon)} = \frac{\mathbf{E}[v_t+\epsilon]^2}{\mathrm{Var}_t(v_t)}$. This leads to the following result for practical BCOSW:

Theorem 4.3. *Suppose Assumptions A and B holds, $\{\alpha_t\}$ satisfies (25) and $\|d_t\|$ is bounded almost surely. Let δ be the smallest constant such that, for all $t \geq 0$,*

$$2c_t\|\sqrt{\rho_t}\| + \mathcal{O}(\epsilon) + \mathcal{O}(\mathrm{Var}_t(v_t)) \leq \lambda\delta. \tag{29}$$

Then we have $\limsup_{t\to\infty} \|x_t - x_*\|^2 \leq \delta^2$, meaning a.s. convergence to a neighborhood of x_*.

In fact, it is sufficient for $\lambda\delta$ to be the $\limsup_{\to\infty}$ of the left-hand side of (29) (see Appendix D.2). We notice from (28) that c_t is small if the estimator v_t has low bias (small τ) and low variance (high SNR). In addition, it also helps to have high SNR of d_t, for example, by using m_t rather than g_t.

Let's examine the bias-variance trade-off of the effective estimator v_t used by popular optimizers:

- The classical SGD method (with $d_t = g_t$ or $d_t = m_t$) effectively uses a constant v_t, which has zero variance but high bias $|\mathbf{E}_t[v_t] - \mathbf{E}_t[d_t^2]| = |v - \mathbf{E}_t[d_t^2]|$ for some constant v.
- Sign-SGD effectively uses $v_t = d_t^2$, which has no bias but high variance $\mathrm{Var}_t(v_t) = \mathrm{Var}_t(d_t)$.
- The conditional estimator of BCOS-c has the following bias and variance (see Appendix E)
$$\mathbf{E}_t[v_t] - \mathbf{E}_t[d_t^2] = 2\beta(1-\beta)m_{t-1}\big(m_{t-1} - \mathbf{E}_t[g_t]\big), \qquad \mathrm{Var}_t(v_t) = (1-\beta)^4\mathrm{Var}_t(g_t^2).$$
Its bias is a small fraction of the bias of m_{t-1} and it has a very small variance.
- For Adam, we do not have a simple expression for its bias, but $\mathrm{Var}_t(v_t) = (1-\beta_2)^2\mathrm{Var}_t(m_t^2)$.

In summary, our convergence analysis can be applied to a variety of different optimizers, including Adam and AdamW, by characterizing their bias-variance trade-off (see Appendix E).

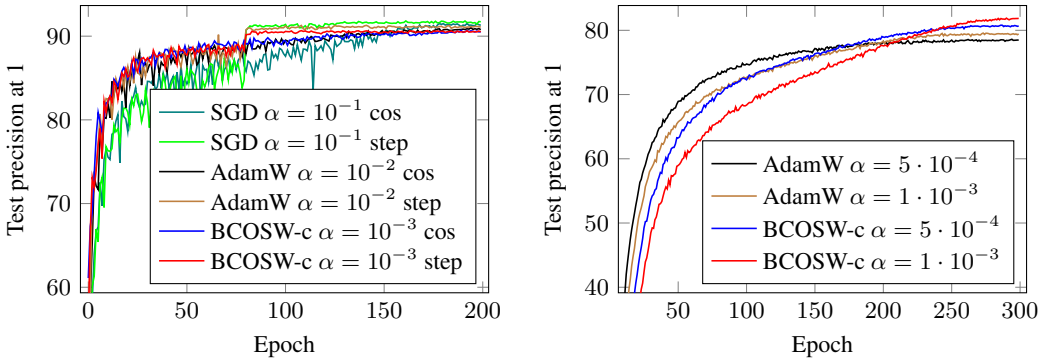

Figure 4: Left: ResNet-20 on CIFAR10. Right: Vision Transformer on ImageNet.

5 Numerical experiments

We present preliminary experiments to compare BCOS with Adam, specifically their variants with decoupled weight decay. Among the BCOSW family, we focus on BCOSW-c (Algorithm 4).

Our first set of experiments are conducted on training the small GPT2 model with 124 million parameters [36] on the OpenWebText dataset [16]. We use global batch size 512 and run all experiments for 100k iterations with the first 2k for linear warmup and then cosine decay on $\{\alpha_t\}$. The default hyper-parameters are chosen (based on a coarse sweep) as: peak stepsize $\alpha_{\max} = 0.002$, final stepsize $\alpha_{\min} = 0.01\alpha_{\max}$, $\epsilon = 10^{-6}$ and weight decay $\lambda = 0.1$.

Figure 1 (left) shows the test loss of AdamW with different combination of β_1 and β_2. For each value of $\beta_1 \in \{0.8, 0.9, 0.95\}$, we choose the best β_2 after sweeping $\beta_2 \in \{0.8, 0.9, 0.95, 0.975, 0.99\}$. Their final loss achieved are all very close around 2.82. For most (β_1, β_2) combinations, we observe loss spikes, especially at the beginning of the training (as shown in the inset). In contrast, Figure 1 (right) shows that BCOSW-c obtains the same final loss but with very smooth loss curve.

Figure 2 compares the test loss of AdamW against the three variants BCOSW-g, -m, and -c. We observe that BCOSW-g is significantly worse than the momentum-based methods. The loss curves for the momentum-based methods are all very close, but with spikes for both AdamW and BCOSW-m.

Figure 3 illustrates the difference between algoritms with and without decoupled weight decay. BCOS-c converges to much higher loss than BCOSW-c, and different values of λ (weight decay) makes dramatic difference for BCOS-c but cause little change to BCOSW-c. The same phenomenon happens for Adam versus AdamW, and we again observe spikes from their loss curve.

Finally, in Figure 4, we compare different algorithms for training ResNet-20 [20] on the CIFAR10 dataset [27], and also training the Vision Transformer (ViT) [50] on the ImageNet dataset [11]. For the ResNet task, we tried both cosine decay (drop by factor 100) and step decay (drop by 10 at epochs 80, 120, 150). The hyper-parameters chosen are: $\beta = 0.9$ for SGD and BCOSW-c, and $\beta_{1,2} = (0.9, 0.99)$ for AdamW. We observe that the best-performing stepsize schedules are quite different for different methods. This prompt the need of tuning hyper-parameters for BCOSW for different tasks even though it shares similar tuned hyper-parameters as AdamW on the GPT2 task.

For the ViT task, although the best tuned stepsize schedules are similar between AdamW and BCOSW, their training and test curves look quite different. Figure 4 (right) shows that the test precision curves for BCOSW-c raises slowly but reaches slightly higher precision at the end.

These preliminary experiments demonstrate that BCOSW-c can obtain competitive performance compared with the state-of-the-art method AdamW, but with fewer optimizer states and fewer hyper-parameters to tune. We are conducting additional empirical study to fully understand its potential.

6 Conclusion

BCOS is a stochastic approximation method that exploits the flexibility of taking different coordinate-wise stepsizes. Rather than using sophisticated ideas from optimization such as preconditioning, it builds upon the simple idea of coordinate-wise contraction and focuses on constructing efficient statistical estimators, especially through conditional expectation, in determining the stepsizes.

References

[1] L. Balles and P. Hennig. Dissecting adam: The sign, magnitude and variance of stochastic gradients. In *International Conference on Machine Learning*, pages 404–413. PMLR, 2018.

[2] J. Bernstein, Y.-X. Wang, K. Azizzadenesheli, and A. Anandkumar. signsgd: Compressed optimisation for non-convex problems. In *International Conference on Machine Learning*, pages 560–569. PMLR, 2018.

[3] J. R. Blum. Multidimensional Stochastic Approximation Methods. *The Annals of Mathematical Statistics*, 25(4):737 – 744, 1954.

[4] S. Bock, J. Goppold, and M. Weiß. An improvement of the convergence proof of the adam-optimizer. *arXiv preprint arXiv:1804.10587*, 2018.

[5] X. Chen, C. Liang, D. Huang, E. Real, K. Wang, H. Pham, X. Dong, T. Luong, C.-J. Hsieh, Y. Lu, et al. Symbolic discovery of optimization algorithms. *Advances in neural information processing systems*, 36:49205–49233, 2023.

[6] K. L. Chung. On a stochastic approximation method. *The Annals of Mathematical Statistics*, pages 463–483, 1954.

[7] K. L. Chung. On a stochastic approximation method. *The Annals of Mathematical Statistics*, pages 463–483, 1954.

[8] A. Defazio, A. Cutkosky, H. Mehta, and K. Mishchenko. Optimal linear decay learning rate schedules and further refinements, 2024.

[9] A. Défossez, L. Bottou, F. Bach, and N. Usunier. A simple convergence proof of adam and adagrad. *arXiv preprint arXiv:2003.02395*, 2020.

[10] B. Delyon and A. Juditsky. Accelerated stochastic approximation. *SIAM Journal on Optimization*, 3(4):868–881, 1993.

[11] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 248–255, 2009.

[12] C. Derman and J. Sacks. On Dvoretzky's Stochastic Approximation Theorem. *The Annals of Mathematical Statistics*, 30(2):601 – 606, 1959.

[13] E. Dinan, S. Yaida, and S. Zhang. Effective theory of transformers at initialization. *arXiv preprint arXiv:2304.02034*, 2023.

[14] J. Duchi, E. Hazan, and Y. Singer. Adaptive subgradient methods for online learning and stochastic optimization. *Journal of Machine Learning Research*, 12(Jul):2121–2159, 2011.

[15] A. Dvoretzky. On stochastic approximation. In *Proceedings of the Third Berkeley Symposium on Mathematical Statistics and Probability*, volume 1, pages 39–55. University of California Press, 1956.

[16] A. Gokaslan and V. Cohen. Openwebtext corpus. `http://Skylion007.github.io/OpenWebTextCorpus`, 2019.

[17] D. M. Gomes, Y. Zhang, E. Belilovsky, G. Wolf, and M. S. Hosseini. Adafisher: Adaptive second order optimization via fisher information. *arXiv preprint arXiv:2405.16397*, 2024.

[18] A. M. Gupal and L. T. Bazhenov. A stochastic analog of the conjugate gradient method. *Cybernetics*, 8(1):138–140, 1972.

[19] K. He, X. Zhang, S. Ren, and J. Sun. Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In *Proceedings of the IEEE international conference on computer vision*, pages 1026–1034, 2015.

[20] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 770–778, 2016.

[21] D. Hwang. Fadam: Adam is a natural gradient optimizer using diagonal empirical fisher information. *arXiv preprint arXiv:2405.12807*, 2024.

[22] R. A. Jacobs. Increased rates of convergence through learning rate adaption. *Neural Networks*, 1:295–307, 1988.

[23] W. Jiang, S. Yang, W. Yang, and L. Zhang. Efficient sign-based optimization: Accelerating convergence via variance reduction. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024.

[24] K. Jordan, Y. Jin, V. Boza, J. You, F. Cesista, L. Newhouse, and J. Bernstein. Muon: An optimizer for hidden layers in neural networks, 2024.

[25] H. Kesten. Accelerated stochastic approximation. *Annals of Mathematical Statistics*, 29(1):41–59, 1958.

[26] D. P. Kingma and J. Ba. Adam: A method for stochastic optimization. In *Proceedings of International Conference on Learning Representations (ICLR)*, 2015. arXiv:1412.6980.

[27] A. Krizhevsky and G. Hinton. Learning multiple layers of features from tiny images. Technical Report 0, University of Toronto, Toronto, Ontario, 2009.

[28] F. Kunstner, A. Milligan, R. Yadav, M. Schmidt, and A. Bietti. Heavy-tailed class imbalance and why adam outperforms gradient descent on language models. *Advances in Neural Information Processing Systems*, 37:30106–30148, 2024.

[29] W. Lin, F. Dangel, R. Eschenhagen, J. Bae, R. E. Turner, and A. Makhzani. Can we remove the square-root in adaptive gradient methods? a second-order perspective. *arXiv preprint arXiv:2402.03496*, 2024.

[30] I. Loshchilov and F. Hutter. Sgdr: Stochastic gradient descent with warm restarts. *arXiv preprint arXiv:1608.03983*, 2016.

[31] I. Loshchilov and F. Hutter. Decoupled weight decay regularization. In *International Conference on Learning Representations (ICLR)*, 2019.

[32] A. R. Mahmood, R. S. Sutton, T. Degris, and P. M. Pilarski. Tuning-free step-size adaption. In *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 2121–2124, 2012.

[33] F. Mirzoakhmedov and S. P. Uryasev. Adaptive step adjustment for a stochastic optimization algorithm. *Zh. Vychisl. Mat. Mat. Fiz.*, 23(6):1314–1325, 1983. [U.S.S.R. Comput. Math. Math. Phys. 23:6, 1983].

[34] B. T. Polyak. Comparison of the rates of convergence of one-step and multi-step optimization algorithms in the presence of noise. *Engineering Cybernetics*, 15:6–10, 1977.

[35] B. T. Polyak and Y. Z. Tsypkin. Pseudogradient adaptation and training algorithms. *Automation and Remote Control,* a translation of *Avtomatika i Telemekhanika*, 34(3):377–397, 1973.

[36] A. Radford, J. Wu, R. Child, D. Luan, D. Amodei, and I. Sutskever. Language models are unsupervised multitask learners. OpenAI Tech Report, 2019.

[37] S. J. Reddi, S. Kale, and S. Kumar. On the convergence of adam and beyond. *arXiv preprint arXiv:1904.09237*, 2019.

[38] H. Robbins and S. Monro. A stochastic approximation method. *The Annals of Mathematical Statistics*, 22(3):400–407, 1951.

[39] H. Robbins and D. Siegmund. A convergence theorem for non negative almost supermartingales and some applications. In J. S. Rustagi, editor, *Optimizing Methods in Statistics*, pages 233–257. Academic Press, 1971.

[40] A. Ruszczyński and W. Syski. Stochastic approximation method with gradient averaging for unconstrained problems. *IEEE Transactions on Automatic Control*, 28(12):1097–1105, 1983.

[41] A. Ruszczyński and W. Syski. Stochastic approximation algorithm with gradient averaging and on-line stepsize rules. In J. Gertler and L. Keviczky, editors, *Proceedings of 9th IFAC World Congress*, pages 1023–1027, Budapest, Hungary, 1984.

[42] A. Ruszczyński and W. Syski. A method of aggregate stochastic subgradients with on-line stepsize rules for convex stochastic programming problems. *Mathematical Programming Study*, 28:113–131, 1986.

[43] A. Ruszczyński and W. Syski. On convergence of the stochastic subgradient method with on-line stepsize rules. *Journal of Mathematical Analysis and Applications*, 114:512–527, 1986.

[44] J. Sacks. Asymptotic distribution of stochastic approximation procedures. *The Annals of Mathematical Statistics*, 29(2):373–405, 1958.

[45] M. Safaryan and P. Richtarik. Stochastic sign descent methods: New algorithms and better theory. In M. Meila and T. Zhang, editors, *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pages 9224–9234. PMLR, 18–24 Jul 2021.

[46] N. N. Schraudolph. Local gain adaptation in stochastic gradient descent. In *Proceedings of Nineth International Conference on Artificial Neural Networks (ICANN)*, pages 569–574, 1999.

[47] R. S. Sutton. Adapting bias by gradient descent: An incremental version of Delta-Bar-Delta. In *Proceedings of the Tenth National Conference on Artificial Intelligence (AAAI'92)*, pages 171–176. The MIT Press, 1992.

[48] T. Tieleman. Lecture 6.5-rmsprop: Divide the gradient by a running average of its recent magnitude. *COURSERA: Neural networks for machine learning*, 4(2):26, 2012.

[49] T. Tieleman and G. Hinton. Lecture 6.5-rmsprop: Divide the gradient by a running average of its recent magnitude. *COURSERA: Neural networks for machine learning*, 4(2):26–31, 2012.

[50] H. Touvron, M. Cord, M. Douze, F. Massa, A. Sablayrolles, and H. Jegou. Training data-efficient image transformers and distillation through attention. In M. Meila and T. Zhang, editors, *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pages 10347–10357. PMLR, 18–24 Jul 2021.

[51] J. H. Venter. On Dvoretzky Stochastic Approximation Theorems. *The Annals of Mathematical Statistics*, 37(6):1534 – 1544, 1966.

[52] M. T. Wasan. *Stochastic Approximation*. Cambridge University Press, 1969.

[53] Z. Yao, A. Gholami, S. Shen, M. Mustafa, K. Keutzer, and M. Mahoney. Adahessian: An adaptive second order optimizer for machine learning. In *proceedings of the AAAI conference on artificial intelligence*, volume 35, pages 10665–10673, 2021.

[54] Y. Zhang, C. Chen, Z. Li, T. Ding, C. Wu, D. P. Kingma, Y. Ye, Z.-Q. Luo, and R. Sun. Adam-mini: Use fewer learning rates to gain more. *arXiv preprint arXiv:2406.16793*, 2024.

[55] Y. Zhang, C. Chen, N. Shi, R. Sun, and Z.-Q. Luo. Adam can converge without any modification on update rules. In S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, and A. Oh, editors, *Advances in Neural Information Processing Systems*, volume 35, pages 28386–28399. Curran Associates, Inc., 2022.

[56] F. Zou, L. Shen, Z. Jie, W. Zhang, and W. Liu. A sufficient condition for convergences of adam and rmsprop. In *Proceedings of the IEEE/CVF Conference on computer vision and pattern recognition*, pages 11127–11135, 2019.

A PyTorch implementation of BCOS

Listing 1: BCOS and BCOSW implementation as a single PyTorch Optimizer

```python
import torch
from torch.optim import Optimizer

class BCOS_short(Optimizer):
    def __init__(self, params, lr, beta=0.9, eps=1e-6,
                 weight_decay=0.1, mode='c', decouple_wd=True):

        defaults = dict(lr=lr, beta=beta, eps=eps, wd=weight_decay)
        super().__init__(params, defaults)

        if mode not in ['g', 'm', 'c']:
            raise ValueError(f"BCOS mode {mode} not supported")
        self.mode = mode
        self.decouple_wd = decouple_wd      # True for BCOSW

    def step(self, closure = None):

        for group in self.param_groups:
            lr = group["lr"]
            beta = group["beta"]
            eps = group["eps"]
            wd = group["wd"]

            for p in group["params"]:
                if not p.requires_grad:
                    continue

                state = self.state[p]
                g = p.grad

                # initialize optimizer states for specific modes
                if self.mode in ['m', 'c'] and 'm' not in state:
                    state['m'] = g.detach().clone()
                if self.mode in ['g', 'm'] and 'v' not in state:
                    state['v'] = g.detach().square()

                # decoupled weight decay or absorb in gradient
                if self.decouple_wd:    # p := (1 - lr * wd) * p
                    p.data.mul_(1 - lr * wd)
                else:                   # g := g + wd * p
                    g.data.add_(p.data, alpha = wd)

                if self.mode in ['m', 'c']:
                    m = state['m']
                    if self.mode == 'c':
                        beta_v = 1 - (1 - beta)**2
                        g2 = g.detach().square()
                        v = beta_v * m.square() + (1 - beta_v) * g2
                    # update momentum
                    m.mul_(beta).add_(g.detach(), alpha=1 - beta)
                    d = m
                else:
                    d = g.detach()

                if self.mode in ['g', 'm']:     # EMA estimator
                    v = state['v']
                    v.mul_(beta).add_(d.square(), alpha=1 - beta)

                # BCOS update: p := p - lr * (d / (sqrt(v) + eps))
                p.data.add_(d.div(v.sqrt() + eps), alpha= - lr)
```

B Aiming condition and convexity

In the paper we have focused on the special case of single coordinate blocks. To investigate the relation between the aiming condition and convexity, it is more instructive to examine the general block structure. For general block partitions $\cup_{k=1}^{m}\mathcal{I}_k$, employing a block-coordinate stepsize vector s_t where each block $\mathcal{I}_k$ of s_t is defined as $s_{t,k} = \widetilde{\gamma}_{t,k}\mathbf{1}_{n_k}$ yields iterative methods of the form

$$x_{t+1} = x_t - s_t \odot d_t, \tag{30}$$

with conceptual BCOS stepsizes

$$\widetilde{\gamma}_{t,k} = \frac{1}{\sqrt{\mathbf{E}_t\left[\|d_{t,k}\|^2\right]}}, \qquad k = 1, \ldots, m.$$

The corresponding aiming condition is as follows, which guarantees one-step improvement.

Assumption C. *There exists $x_* \in \mathbf{R}^n$ such that*

$$\sum_{k=1}^{m}\left\langle x_t - x_*, \frac{\mathbf{E}_t[d_t]}{\sqrt{\mathbf{E}_t\left[\|d_{t,k}\|^2\right]}}\right\rangle \geq 0, \tag{31}$$

holds for all $t \geq 0$ almost surely. If d_t is independent of the past trajectory conditioned on x_t, i.e., $\mathbf{E}_t[d_t] = \mathbf{E}[d_t|x_t]$, then it suffices to have (31) *hold for every $x \in \mathbf{R}^n$.*

Assumption C allows us to conduct a comparative analysis of the aiming condition and the classical convexity assumption, highlighting their similarities and key differences. For the sake of simplicity in our exposition, we will assume that the stochastic search direction d_t is trajectory independent, i.e., $\mathbf{E}_t[d_t] = \mathbf{E}[d_t|x_t]$, allowing us to drop the subscript t. We further assume that d_t satisfies $\mathbf{E}[d] = \nabla f(x)$. Simplifying (31):

$$\sum_{k=1}^{m}\left\langle x_k - x_{*,k}, \frac{\nabla f(x)_k}{\|\nabla f(x)_k\|}\right\rangle \geq 0, \qquad \forall x. \tag{32}$$

In the specific case of a full-dimensional block stepsize, where $\widetilde{\gamma}_t = \frac{1}{\|\nabla f(x)\|} \in \mathbf{R}_+$ is a scaler and the stepsize vector is $s_t = \widetilde{\gamma}_t\mathbf{1}_n$, the aiming condition simplifies to:

$$\langle x - x_*, \nabla f(x)\rangle \geq 0, \qquad \forall x. \tag{33}$$

Condition (33) is directly implied by the classical convex assumption, which states:

$$\langle x - y, \nabla f(x) - \nabla f(y)\rangle \geq 0, \qquad \forall x, y. \tag{34}$$

To see the implication, simply substitute $y = x_*$ and $\nabla f(x_*) = 0$ into the above convex inequality.

However, the aiming condition under a general block partition exhibits a significant departure from the classical notion of convexity, as expected update directions deviate from true gradients and become axis-aligned. Consider the extreme case of coordinate-wise stepsizes, where $s_t = \widetilde{\gamma}_t \in \mathbf{R}^n$ and each element is chosen as $\widetilde{\gamma}_{t,k} = \frac{1}{\sqrt{\nabla f(x_k)^2}} = \frac{1}{|\nabla f(x_k)|}$. The specific choice of stepsize yields an aiming condition of the form:

$$\langle x - x_*, \text{sign}(\nabla f(x))\rangle \geq 0, \qquad \forall x. \tag{35}$$

To illustrate the fundamental differences between this coordinate-wise aiming condition (35) and the standard convexity assumption (34), we provide the following two counterexamples, each satisfying one condition while failing the other:

- *Aiming but not convex*: Let $f(x) := \log(x)$ with the optimal solution $x_* = 0$. On the domain of $\mathbf{R}_+$, the gradient is $f'(x) = \frac{1}{x}$, and thus $\text{sign}(f'(x)) = 1$ for all $x > 0$. Consequently, for any $x \in \mathbf{R}_+$, we have
$$\langle x - x_*, \text{sign}(\nabla f(x))\rangle = x \geq 0,$$
satisfying the aiming condition (35). However, $\log(x)$ is a concave function, thus failing the convex inequality (34).

- *Convex but not aiming*: Consider the quadratic function class $f : \mathbf{R}^2 \to \mathbf{R}$, $f(x) = \frac{1}{2}x^T Ax$. Choose coefficient matrix A:

$$A = \begin{pmatrix} 1 \\ -2 \end{pmatrix} \begin{pmatrix} 1 \\ -2 \end{pmatrix}^T = \begin{bmatrix} 1 & -2 \\ -2 & 4 \end{bmatrix} \succeq 0.$$

Since A is positive semidefinite, the function f is convex and attains its minimum at $x_* = \mathbf{0}$. The gradient of f is

$$\nabla f(x) = Ax = \begin{bmatrix} x_1 - 2x_2 \\ -2x_1 + 4x_2. \end{bmatrix}.$$

Evaluating the aiming condition (35) at $x = (1.5, 1)^T$, we get

$$\langle x - x_*, \text{sign}(\nabla f(x)) \rangle = 1.5 \times \text{sign}(-0.5) + 1 \times \text{sign}(1) = 1.5 \times (-1) + 1 \times 1 = -0.5 \leq 0.$$

Thus, the aiming condition (35) at this point even though f is convex.

C Convergence analysis of conceptual BCOSW

First, we notice that the aiming condition is Assumption A is equivalent to

$$\left\langle x_t - x_*,\ \sqrt{\rho_t} \odot \text{sign}(\mathbf{E}_t[d_t]) + \lambda x_* \right\rangle \geq 0, \tag{36}$$

because

$$\begin{aligned} &\left\langle x_t - x_*,\ \sqrt{\rho_t} \odot \text{sign}(\mathbf{E}_t[d_t]) + \lambda x_* \right\rangle \\ =&\left\langle x_t - x_*,\ \sqrt{\rho_t} \odot \text{sign}(\mathbf{E}_t[d_t]) + \lambda x_t - \lambda x_t + \lambda x_* \right\rangle \\ =&\left\langle x_t - x_*,\ \sqrt{\rho_t} \odot \text{sign}(\mathbf{E}_t[d_t]) + \lambda x_t \right\rangle - \lambda \|x_t - x_*\|^2. \end{aligned}$$

We use it to prove Lemma 4.1.

Proof of Lemma 4.1. Given $x_{t+1} = x_t - \widetilde{\gamma}_t \odot d_t - \alpha_t x_t$, we have

$$\begin{aligned} \mathbf{E}_t[\|x_{t+1} - x_*\|^2] &= \mathbf{E}_t \left[\left\| x_t - \frac{\alpha_t}{\sqrt{\mathbf{E}_t[d_t^2]}} \odot d_t - \alpha_t \lambda x_t - x_* \right\|^2 \right] \\ &= \mathbf{E}_t \left[\left\| (1 - \alpha_t\lambda)x_t - \frac{\alpha_t}{\sqrt{\mathbf{E}_t[d_t^2]}} \odot d_t - (1 - \alpha_t\lambda)x_* - \alpha_t\lambda x_* \right\|^2 \right] \\ &= \mathbf{E}_t \left\| (1 - \alpha_t\lambda)(x_t - x_*) \right\|^2 - 2\mathbf{E}_t \left\langle (1 - \alpha_t\lambda)(x_t - x_*), \frac{\alpha_t}{\sqrt{\mathbf{E}_t[d_t^2]}} \odot d_t + \alpha_t\lambda x_* \right\rangle \\ &\quad + \mathbf{E}_t \left\| \frac{\alpha_t}{\sqrt{\mathbf{E}_t[d_t^2]}} \odot d_t + \alpha_t\lambda x_* \right\|^2 \\ &= (1 - \alpha_t\lambda)^2 \left\| x_t - x_* \right\|^2 - 2\alpha_t(1 - \alpha_t\lambda) \left\langle x_t - x_*,\ \frac{\mathbf{E}_t[d_t]}{\sqrt{\mathbf{E}_t[d_t^2]}} + \lambda x_* \right\rangle \\ &\quad + \alpha_t^2 \sum_k \left(1 + \lambda^2 x_{*,k}^2 + 2\lambda x_{*,k} \frac{\mathbf{E}_t[d_{t,k}]}{\sqrt{\mathbf{E}_t[d_{t,k}^2]}} \right) \\ &= (1 - \alpha_t\lambda)^2 \left\| x_t - x_* \right\|^2 - 2\alpha_t(1 - \alpha_t\lambda) \left\langle x_t - x_*,\ \sqrt{\rho_t}\text{sign}\left(\mathbf{E}_t[d_t]\right) + \lambda x_* \right\rangle \\ &\quad + \alpha_t^2 \sum_k \left(1 + \lambda^2 x_{*,k}^2 + 2\lambda x_{*,k} \sqrt{\rho_{t,k}} \text{sign}\left(\mathbf{E}_t[d_{t,k}]\right) \right) \end{aligned}$$

Under Assumption A, the aiming condition in (36) implies that the inner product in the last equality above is non-negative. With $\alpha_t \geq 0$ and $\alpha_t\lambda \leq 1$, we can drop the inner product term to obtain

$$\begin{aligned} \mathbf{E}_t[\|x_{t+1} - x_*\|^2] &\leq (1 - \alpha_t\lambda)^2 \left\| x_t - x_* \right\|^2 + \alpha_t^2 \sum_k \left(1 + \lambda^2 x_{*,k}^2 + 2\lambda x_{*,k} \sqrt{\rho_{t,k}} \text{sign}\left(\mathbf{E}_t[d_{t,k}]\right) \right) \\ &\leq (1 - \alpha_t\lambda)^2 \left\| x_t - x_* \right\|^2 + \alpha_t^2 (n + \lambda^2 \left\| x_* \right\|^2 + 2\lambda \left\| x_* \right\|_1) \end{aligned}$$

where the last inequality follows from the loose upper bound $\sqrt{\rho_{t,k}}\text{sign}\left(\mathbf{E}_t[d_{t,k}]\right) \leq 1$. □

The proof of Theorem 4.1 follows from the following almost supermartingale lemma.

Lemma C.1 ("Almost supermartingale", Theorem 1 [39]). *) Let $(\Omega, \mathcal{F}, P)$ be a probability space, and $\mathcal{F}_0 \subset \mathcal{F}_1 \subset \ldots$ be a sequence of sub-σ-algebras of $\mathcal{F}$. For each t, let X_t, a_t, b_t, c_t be non-negative $\mathcal{F}_t$-measurable random variables such that*

$$\mathbf{E}[X_{t+1}|\mathcal{F}_t] \leq X_t(1 + a_t) + b_t - c_t. \tag{37}$$

Given $\sum_{t=0}^{\infty} a_t < \infty$ and $\sum_{t=0}^{\infty} b_t < \infty$, then $\lim_{t\to\infty} X_t$ exists and is finite, and $\sum_{t=0}^{\infty} c_t < \infty$ almost surely (a.s.).

Proof of Theorem 4.1. Define $X_t := \|x_t - x_*\|^2$ and $\mathcal{F}_t$ to be the σ-algebra generated by $X_0, \cdots, X_t$. Lemma 4.1 implies the following recursive relationship

$$\begin{aligned}
\mathbf{E}[X_{t+1}|\mathcal{F}_t] &= \mathbf{E}_t[\|x_{t+1} - x_*\|^2] \\
&\leq (1 - \alpha_t\lambda)^2 \|x_t - x_*\|^2 + \alpha_t^2 c_* \\
&= (1 + \alpha_t^2\lambda^2) \|x_t - x_*\|^2 + \alpha_t^2 c_* - 2\alpha_t\lambda \|x_t - x_*\|^2 \\
&= (1 + a_t) X_t + b_t - c_t,
\end{aligned}$$

In the form of (37), we have $a_t = \alpha_t^2\lambda^2$, $b_t = \alpha_t^2 c_*$, $c_t = 2\alpha_t\lambda \|x_t - x_*\|^2$. Here, X_t, a_t, b_t, c_t are trivially non-negative, and the squared summable assumption of α_t guarantees:

$$\sum_{t=0}^{\infty} a_t = \sum_{t=0}^{\infty} \alpha_t^2\lambda^2 < \infty, \qquad \sum_{t=0}^{\infty} b_t = \sum_{t=0}^{\infty} \alpha_t^2 c_* < \infty.$$

So far, we have verified all the assumptions in Lemma C.1, so we conclude that

$$X_t = \|x_t - x_*\|^2 \to X \quad \text{a.s. for some } X < \infty, \qquad \sum_{t=0}^{\infty} c_t = \sum_{t=0}^{\infty} 2\alpha_t\lambda \|x_t - x_*\|^2 < \infty \quad \text{a.s.}$$

This is compatible with $\sum_{t=0}^{\infty} \alpha_t = \infty$ only if

$$\|x_t - x_*\|^2 \to 0 \quad \text{a.s.},$$

as desired. □

To quantify the convergence rate, we study the upper bound on the expected distance to the optimal solution $\mathbf{E}[\|x_T - x_*\|^2]$, after recursively applying BCOSW for T iterations.

Theorem C.1. *Suppose Assumption A holds, $\alpha_t \geq 0$ and $\alpha_t\lambda \leq 1$. The expected distance to x_* admits the following upper bound after T iterations of BCOSW:*

$$\mathbf{E}[\|x_T - x_*\|^2] \leq \prod_{t=0}^{T-1} (1 - \alpha_t\lambda)^2 \mathbf{E}\left[\|x_0 - x_*\|^2\right] + \sum_{t=0}^{T-1} \prod_{t'=t+1}^{T-1} (1 - \alpha_{t'}\lambda)^2 \alpha_t^2 c_*, \tag{38}$$

where $c_* := (n + \lambda^2 \|x_*\|^2 + 2\lambda \|x_*\|_1)$ denote the constant residual that depends on x_*.

Proof. Taking expectation of the recursive relationship (24) and applying the law of total expectation, we obtain:

$$\begin{aligned}
\mathbf{E}[\|x_T - x_*\|^2] &= \mathbf{E}\left[\mathbf{E}_{T-1}[\|x_T - x_*\|^2]\right] \\
&\leq \mathbf{E}\left[(1-\alpha_{T-1}\lambda)^2\|x_{T-1}-x_*\|^2 + \alpha_{T-1}^2 c_*\right] \\
&= (1-\alpha_{T-1}\lambda)^2\mathbf{E}\left[\|x_{T-1}-x_*\|^2\right] + \alpha_{T-1}^2 c_* \\
&= (1-\alpha_{T-1}\lambda)^2\mathbf{E}\left[\mathbf{E}_{T-1}[\|x_{T-1}-x_*\|^2]\right] + \alpha_{T-1}^2 c_* \\
&\leq (1-\alpha_{T-1}\lambda)^2\mathbf{E}\left[(1-\alpha_{T-2}\lambda)^2\|x_{T-2}-x_*\|^2 + \alpha_{T-2}^2 c_*\right] + \alpha_{T-1}^2 c_* \\
&= (1-\alpha_{T-1}\lambda)^2(1-\alpha_{T-2}\lambda)^2\mathbf{E}\left[\|x_{T-2}-x_*\|^2\right] + ((1-\alpha_{T-1}\lambda)^2\alpha_{T-2}^2 + \alpha_{T-1}^2)c_* \\
&\vdots \\
&\leq \prod_{t=0}^{T-1}(1-\alpha_t\lambda)^2\mathbf{E}\left[\|x_0-x_*\|^2\right] + \sum_{t=0}^{T-1}\prod_{t'=t+1}^{T-1}(1-\alpha_{t'}\lambda)^2\alpha_t^2 c_*,
\end{aligned}$$

as desired. □

Different choices of stepsize schedule lead to different convergence behaviors. Next, we consider two choices of α_t: (i) diminishing learning rates $\alpha_t = \frac{\alpha}{t+1}$, which leads to Theorem 4.2 and (ii) constant learning rates $\alpha_t = \alpha$ which lead to linear convergence to a neighborhood of x_*.

The proof of Theorem 4.2 is a direct application of a classical result in the 1954 paper of Chung's [7].

Lemma C.2 (Chung's lemma, Lemma 1 from [7])**.** *Suppose that $\{X_t\}$ is a sequence of real numbers such that for t,*

$$X_{t+1} \leq \left(1 - \frac{a}{t}\right) X_t + \frac{b}{t^{p+1}}, \tag{39}$$

where $a > p > 0, b > 0$. Then

$$X_t \leq \frac{b}{a-p}\frac{1}{t^p} + \mathcal{O}\left(\frac{1}{t^{p+1}} + \frac{1}{t^a}\right).$$

Proof of Theorem 4.2. Taking expectation of both sides of (24) with $\alpha_t = \frac{\alpha}{t+1}$ at iteration T, we have

$$\begin{aligned}
&\mathbf{E}[\|x_T - x_*\|^2] \\
&\leq (1-\alpha_{T-1}\lambda)^2\mathbf{E}[\|x_{T-1}-x_*\|^2] + \alpha_{T-1}^2 c_* \\
&= \left(1 - \frac{\alpha\lambda}{T}\right)^2 \mathbf{E}[\|x_{T-1}-x_*\|^2] + \frac{\alpha^2}{T^2}c_* \\
&= \left(1 - \frac{2\alpha\lambda}{T}\right)\mathbf{E}[\|x_{T-1}-x_*\|^2] + \frac{\alpha^2}{T^2}\left(c_* + \lambda^2\mathbf{E}[\|x_{T-1}-x_*\|^2]\right) \\
&\leq \left(1 - \frac{2\alpha\lambda}{T}\right)\mathbf{E}[\|x_{T-1}-x_*\|^2] + \frac{\alpha^2 c_*}{T^2} \\
&+ \frac{\alpha^2\lambda^2}{T^2}\left(\prod_{t=0}^{T-2}\left(1 - \frac{\alpha\lambda}{t+1}\right)^2 \mathbf{E}\left[\|x_0-x_*\|^2\right] + \sum_{t=0}^{T-2}\prod_{t'=t+1}^{T-2}\left(1 - \frac{\alpha\lambda}{t'+1}\right)^2 \frac{\alpha^2 c_*}{(t+1)^2}\right),
\end{aligned}$$

where the last inequality is in light of (38) in Theorem C.1 and $\alpha_t = \frac{\alpha}{t+1}$. Upper bounding $\left(1 - \frac{\alpha\lambda}{t+1}\right)^2$ by 1 yields:

$$\mathbf{E}[\|x_T - x_*\|^2] \leq \left(1 - \frac{2\alpha\lambda}{T}\right)\mathbf{E}[\|x_{T-1}-x_*\|^2] + \frac{\alpha^2 c_*}{T^2} + \frac{\alpha^2\lambda^2}{T^2}\left(\mathbf{E}\left[\|x_0-x_*\|^2\right] + \sum_{t=0}^{T-2}\frac{\alpha^2 c_*}{(t+1)^2}\right).$$

Further replacing the finite sum $\sum_{t=0}^{T-2} \frac{1}{(t+1)^2} = \sum_{t=1}^{T-1} \frac{1}{t^2}$ by its infinite version $\frac{\pi^2}{6}$, we obtain a recursive relationship in the form of (39):

$$\mathbf{E}[\|x_T - x_*\|^2] \leq \left(1 - \frac{2\alpha\lambda}{T}\right) \mathbf{E}[\|x_{T-1} - x_*\|^2] + \frac{\alpha^2 c_*}{T^2} + \frac{\alpha^2\lambda^2}{T^2} \left(\mathbf{E}\left[\|x_0 - x_*\|^2\right] + \frac{\pi^2\alpha^2 c_*}{6}\right),$$

with $X_t = \mathbf{E}[\|x_{t-1} - x_*\|^2], a = 2\alpha\lambda, b = \alpha^2 c_* + \alpha^2\lambda^2 \left(\mathbf{E}\left[\|x_0 - x_*\|^2\right] + \frac{\pi^2\alpha^2 c_*}{6}\right)$, and $p = 1$, which satisfies the Chung's assumptions $a > 1 = p > 0, b > 0$ because $\alpha\lambda \in (0.5, 1)$. Lemma C.2 implies

$$\mathbf{E}[\|x_T - x_*\|^2] \leq \frac{\alpha^2 c_* + \alpha^2\lambda^2 \left(\mathbf{E}\left[\|x_0 - x_*\|^2\right] + \frac{\pi^2\alpha^2 c_*}{6}\right)}{2\alpha\lambda - 1} \frac{1}{T} + \mathcal{O}\left(\frac{1}{T^2} + \frac{1}{T^{2\alpha\lambda}}\right),$$

as desired. □

With a constant stepsize, we obtain linear convergence to a neighborhood of x_*, as stated in the following corollary.

Corollary C.2. *Fix learning rate schedule $\alpha_t = \alpha$ where α satisfies $\alpha\lambda < 1$. Let x_t's be a sequence generated by applying the conceptual BCOSW. Under Assumption A, the asymptotic expected distance to x_* admits the following upper bound:*

$$\mathbf{E}[\|x_T - x_*\|^2] \leq (1 - \alpha\lambda)^{2T} \mathbf{E}\left[\|x_0 - x_*\|^2\right] + \frac{\alpha^2 c_*}{1 - (1 - \alpha\lambda)^2}. \tag{40}$$

Proof. A direct application of Theorem C.1 with $\alpha_t = \alpha$ yields the following upper bound on: $\mathbf{E}[\|x_T - x_*\|^2]$

$$\begin{aligned}
\mathbf{E}[\|x_T - x_*\|^2] &\leq (1 - \alpha\lambda)^{2T} \mathbf{E}\left[\|x_0 - x_*\|^2\right] + \sum_{t=0}^{T-1} (1 - \alpha\lambda)^{2(T-t-1)} \alpha^2 c_* \\
&= (1 - \alpha\lambda)^{2T} \mathbf{E}\left[\|x_0 - x_*\|^2\right] + \sum_{t=0}^{T-1} (1 - \alpha\lambda)^{2t} \alpha^2 c_* \\
&\leq (1 - \alpha\lambda)^{2T} \mathbf{E}\left[\|x_0 - x_*\|^2\right] + \frac{\alpha^2 c_*}{1 - (1 - \alpha\lambda)^2},
\end{aligned}$$

which decreases exponentially with T and converges to a constant. □

D Convergence analysis of practical BCOSW

D.1 Proof of Lemma 4.2

We first prove Lemma 4.2. To proceed, we decompose the error between the expected search directions into two parts (elementwise inequality between vectors):

$$\left|\frac{\mathbf{E}_t[d_t]}{\sqrt{\mathbf{E}[d_t^2]}} - \mathbf{E}_t\left[\frac{d_t}{\sqrt{v_t + \epsilon}}\right]\right| \leq \left|\frac{\mathbf{E}_t[d_t]}{\sqrt{\mathbf{E}[d_t^2]}} - \frac{\mathbf{E}_t[d_t]}{\sqrt{\mathbf{E}_t[v_t] + \epsilon}}\right| + \left|\frac{\mathbf{E}_t[d_t]}{\sqrt{\mathbf{E}_t[v_t] + \epsilon}} - \mathbf{E}_t\left[\frac{d_t}{\sqrt{v_t + \epsilon}}\right]\right|. \tag{41}$$

Under certain assumptions on the quality of the estimator v_t, we demonstrate that the practical update approximates the conceptual update in expectation by bounding the two terms on the right-hand side separately.

Assumption B leads to an upper bound for the first error term in (41).

Lemma D.1. *Under Assumption B, it holds that:*

$$\left|\frac{\mathbf{E}_t[d_t]}{\sqrt{\mathbf{E}_t[d_t^2]}} - \frac{\mathbf{E}_t[d_t]}{\sqrt{\mathbf{E}_t[v_t] + \epsilon}}\right| \leq \frac{4\tau + 3\tau^2}{8} \left|\frac{\mathbf{E}_t[d_t]}{\sqrt{\mathbf{E}_t[d_t^2]}}\right| + \mathcal{O}(\epsilon). \tag{42}$$

Proof. The proof leverages the second-order Taylor expansion of $g(y) := \frac{1}{\sqrt{y}}$:

$$g(y+\delta) \approx g(y) + g'(y)\delta + \frac{1}{2}g''(y)\delta^2, \qquad \text{where} \quad g'(y) = -\frac{1}{2y^{3/2}}, \quad g''(y) = \frac{3}{4y^{5/2}}.$$

Applying Taylor expansion at $y := \mathbf{E}_t[d_t^2]$ with $\delta := \mathbf{E}_t[v_t] + \epsilon - \mathbf{E}_t[d_t^2]$ yields the following approximation:

$$\begin{aligned}
&\left|\frac{\mathbf{E}_t[d_t]}{\sqrt{\mathbf{E}_t[d_t^2]}} - \frac{\mathbf{E}_t[d_t]}{\sqrt{\mathbf{E}_t[v_t]+\epsilon}}\right| \\
=&\left|\mathbf{E}_t[d_t](g(y) - g(y+\delta))\right| \\
\approx&\left|\mathbf{E}_t[d_t](g'(y)\delta + \frac{1}{2}g''(y)\delta^2 + \mathcal{O}(\delta^3))\right| \\
=&\left|\mathbf{E}_t[d_t]\left(-\frac{1}{2\mathbf{E}_t[d_t^2]^{3/2}}\cdot(\mathbf{E}_t[v_t]+\epsilon-\mathbf{E}_t[d_t^2]) + \frac{3}{8\mathbf{E}_t[d_t^2]^{5/2}}\cdot(\mathbf{E}_t[v_t]+\epsilon-\mathbf{E}_t[d_t^2])^2\right)\right| + \mathcal{O}(\epsilon) \\
\le&\left|\mathbf{E}_t[d_t]\left(\frac{1}{2\mathbf{E}_t[d_t^2]^{3/2}}\cdot\tau\mathbf{E}_t[d_t^2] + \frac{3}{8\mathbf{E}_t[d_t^2]^{5/2}}\cdot\tau^2\mathbf{E}_t[d_t^2]^2\right)\right| + \mathcal{O}(\epsilon) \qquad (43) \\
\le&\frac{4\tau+3\tau^2}{8}\left|\frac{\mathbf{E}_t[d_t]}{\sqrt{\mathbf{E}_t[d_t^2]}}\right| + \mathcal{O}(\epsilon),
\end{aligned}$$

where (43) is a consequence of Assumption B. □

To establish the upper bound on the second error term in (41), $\left|\frac{\mathbf{E}_t[d_t]}{\sqrt{\mathbf{E}_t[v_t]+\epsilon}} - \mathbf{E}_t\left[\frac{d_t}{\sqrt{v_t+\epsilon}}\right]\right|$, we present a useful approximation for general differential function g.

Lemma D.2. *For any differentiable function g and random variable $X \in \mathbf{R}^n$, the following expansion holds:*

$$\mathbf{E}\left[g(X)\right] = g(\mathbf{E}[X]) + \frac{1}{2}\langle\nabla^2 g(\mathbf{E}[X]), \mathrm{Cov}(X)\rangle + \sum_{p=3}^{\infty}\frac{D^p g(\mathbf{E}[X])}{p!}\mathbf{E}\left[(X-\mathbf{E}[X])^p\right], \qquad (44)$$

where $\langle\cdot,\cdot\rangle$ denotes matrix inner product, i.e, $\langle A, B\rangle = \mathrm{Tr}(A^T B)$, and $p \in \mathbf{N}^n$ and

$$D^p g(\mathbf{E}[X]) = \frac{\partial^{|p|} g}{\partial X^p} = \frac{\partial^{p_1+\cdots+p_n} g}{\partial X_1^{p_1}\cdots\partial X_n^{p_n}}.$$

Proof. Let $\delta := X - \mathbf{E}[X]$. The second-order Taylor expansion of g at $\mathbf{E}[X]$ yields

$$g(X) = g(\mathbf{E}[X]) + \nabla g(\mathbf{E}[X])^T\delta + \frac{1}{2}\delta^T\nabla^2 g(\mathbf{E}[X])\delta + \sum_{p=3}^{\infty}\frac{D^p g(\mathbf{E}[X])}{p!}\delta^p.$$

Taking expectation with respect to X, we have

$$\begin{aligned}
\mathbf{E}[g(X)] &= g(\mathbf{E}[X]) + \nabla g(\mathbf{E}[X])^T\mathbf{E}[\delta] + \frac{1}{2}\mathbf{E}[\delta^T\nabla^2 g(\mathbf{E}[X])\delta] + \sum_{p=3}^{\infty}\frac{D^p g(\mathbf{E}[X])}{p!}\mathbf{E}\left[\delta^p\right] \\
&= g(\mathbf{E}[X]) + 0 + \frac{1}{2}\left\langle\nabla^2 g(\mathbf{E}[X]),\, \mathbf{E}[\delta\delta^T]\right\rangle + \sum_{p=3}^{\infty}\frac{D^p g(\mathbf{E}[X])}{p!}\mathbf{E}\left[\delta^p\right],
\end{aligned}$$

where $\mathbf{E}[\delta\delta^T] = \mathrm{Cov}(X)$ and $\mathbf{E}\left[\delta^p\right] = \mathbf{E}\left[(X-\mathbf{E}[X])^p\right]$. □

The following lemma provides an approximation for $\mathbf{E}[g]$ with $g(Y,Z) := \frac{Y}{\sqrt{Z}}$.

Lemma D.3. *Let Y, Z be two random variables and $Z > 0$ almost surely, then*

$$\begin{aligned}
\mathbf{E}\left[\frac{Y}{\sqrt{Z}}\right] =& \frac{\mathbf{E}[Y]}{\sqrt{\mathbf{E}[Z]}}\left(1 - \frac{\mathrm{Cov}(Y,Z)}{2\mathbf{E}[Y]\mathbf{E}[Z]} + \frac{3\mathrm{Var}(Z)}{8\mathbf{E}[Z]^2}\right) \\
&+ \mathcal{O}\left(\mathbf{E}[(Y-\mathbf{E}[Y])(Z-\mathbf{E}[Z])^2]\right) + \mathcal{O}\left(\mathbf{E}[(Z-\mathbf{E}[Z])^3]\right).
\end{aligned} \qquad (45)$$

Proof. We apply Lemma D.2 with $X := (Y, Z)$ and $g(x) = g(y, z) := \frac{y}{\sqrt{z}}$. First, the gradient and Hessian of g can be calculated as

$$\nabla g(x) = \nabla g(y,z) = \begin{pmatrix} \frac{1}{z^{1/2}} \\ -\frac{y}{2z^{3/2}} \end{pmatrix}, \qquad \nabla^2 g(x) = \nabla^2 g(y,z) = \begin{bmatrix} 0, & -\frac{1}{2z^{3/2}} \\ -\frac{1}{2z^{3/2}}, & \frac{3y}{4z^{5/2}} \end{bmatrix}.$$

For general p-th partial derivative, we derive the following result for any $q \in [0, p]$:

$$\frac{\partial^p g}{\partial y^q \partial z^{p-q}} = \frac{\partial^{p-q}}{\partial z^{p-q}} \left(\frac{\partial^q g}{\partial y^q} \right) = \begin{cases} 0 & \text{if } q \geq 2, \\ \frac{\partial^{p-1}}{\partial z^{p-1}} \frac{1}{\sqrt{z}} = (-1)^{p-1} \frac{(2p-2)!}{4^{p-1}(p-1)!} z^{-\frac{2p-1}{2}} & \text{if } q = 1, \\ y \cdot \frac{\partial^p}{\partial z^p} \frac{1}{\sqrt{z}} = (-1)^p \frac{(2p)!}{4^p p!} y z^{-\frac{2p+1}{2}} & \text{if } q = 0. \end{cases},$$

which Substitute the gradient, Hessian and p-th order partial derivative into (44), we get

$$\begin{aligned}
&\mathbf{E}\left[\frac{Y}{\sqrt{Z}}\right] \\
&= \frac{\mathbf{E}[Y]}{\sqrt{\mathbf{E}[Z]}} - \mathbf{E}\left[\frac{(Y - \mathbf{E}[Y])(Z - \mathbf{E}[Z])}{2\mathbf{E}[Z]^{3/2}}\right] + \mathbf{E}\left[\frac{3\mathbf{E}[Y](Z - \mathbf{E}[Z])^2}{8\mathbf{E}[Z]^{5/2}}\right] \\
&+ \sum_{p=3}^{\infty} \frac{1}{p!} \left(\frac{p(2p-2)!}{4^{p-1}(p-1)!} \mathbf{E}\left[\frac{(Y - \mathbf{E}[Y])(Z - \mathbf{E}[Z])^{p-1}}{\mathbf{E}[Z]^{\frac{2p-1}{2}}}\right] + (-1)^p \frac{(2p)!}{4^p p!} \mathbf{E}\left[\frac{\mathbf{E}[Y](Z - \mathbf{E}[Z])^p}{\mathbf{E}[Z]^{\frac{2p+1}{2}}}\right] \right) \\
&= \frac{\mathbf{E}[Y]}{\sqrt{\mathbf{E}[Z]}} - \frac{\mathrm{Cov}(Y,Z)}{2\mathbf{E}[Z]^{3/2}} + \frac{3\mathbf{E}[Y]\mathrm{Var}(Z)}{8\mathbf{E}[Z]^{5/2}} + \mathcal{O}\left(\mathbf{E}[(Y - \mathbf{E}[Y])(Z - \mathbf{E}[Z])^2]\right) + \mathcal{O}\left(\mathbf{E}[(Z - \mathbf{E}[Z])^3]\right) \\
&= \frac{\mathbf{E}[Y]}{\sqrt{\mathbf{E}[Z]}} \left(1 - \frac{\mathrm{Cov}(Y,Z)}{2\mathbf{E}[Y]\mathbf{E}[Z]} + \frac{3\mathrm{Var}(Z)}{8\mathbf{E}[Z]^2}\right) + \mathcal{O}\left(\mathbf{E}[(Y - \mathbf{E}[Y])(Z - \mathbf{E}[Z])^2]\right) + \mathcal{O}\left(\mathbf{E}[(Z - \mathbf{E}[Z])^3]\right),
\end{aligned}$$

as desired. □

A combination of the consequence of Lemma D.3 and Assumption B culminates in an upper bound on the second error term.

Lemma D.4. *Define signal-noise-ratio* $\mathrm{SNR}_t(Y) := \frac{\mathbf{E}_t[Y_t]^2}{\mathrm{Var}_t(Y_t)}$. *Under Assumptions B, we have*

$$\begin{aligned}
&\left| \frac{\mathbf{E}_t[d_t]}{\sqrt{\mathbf{E}_t[v_t] + \epsilon}} - \mathbf{E}_t\left[\frac{d_t}{\sqrt{v_t + \epsilon}}\right] \right| \\
&\leq \left(\frac{8 + 4\tau + 3\tau^2}{8}\right) \left| -\frac{\mathrm{Corr}_t(d_t, v_t + \epsilon)}{2\sqrt{\mathrm{SNR}_t(v_t + \epsilon)}} \sqrt{\frac{1}{\rho_t} - 1} + \frac{3}{8\mathrm{SNR}_t(v_t + \epsilon)} \right| \cdot \left| \frac{\mathbf{E}_t[d_t]}{\sqrt{\mathbf{E}_t[d_t^2]}} \right| \\
&\quad + \mathcal{O}(\epsilon) + \mathcal{O}\left(\mathbf{E}_t[(Y - \mathbf{E}_t[Y])(Z - \mathbf{E}_t[Z])^2]\right) + \mathcal{O}\left(\mathbf{E}_t[(Z - \mathbf{E}_t[Z])^3]\right)
\end{aligned}$$

Proof. Following Lemma D.3 with $Y := d_t, Z := v_t + \epsilon$, we get

$$\begin{aligned}
\left| \frac{\mathbf{E}_t[d_t]}{\sqrt{\mathbf{E}_t[v_t] + \epsilon}} - \mathbf{E}_t\left[\frac{d_t}{\sqrt{v_t + \epsilon}}\right] \right| &\leq \left| \frac{\mathbf{E}_t[d_t]}{\sqrt{\mathbf{E}_t[v_t] + \epsilon}} \right| \cdot \left| -\frac{\mathrm{Cov}_t(d_t, v_t + \epsilon)}{2\mathbf{E}_t[d_t]\mathbf{E}_t[v_t + \epsilon]} + \frac{3\mathrm{Var}_t(v_t + \epsilon)}{8\mathbf{E}_t[v_t + \epsilon]^2} \right| \quad (46) \\
&\quad + \mathcal{O}\left(\mathbf{E}_t[(Y - \mathbf{E}_t[Y])(Z - \mathbf{E}_t[Z])^2]\right) + \mathcal{O}\left(\mathbf{E}_t[(Z - \mathbf{E}_t[Z])^3]\right).
\end{aligned}$$

We express the covariance between d and $v + \epsilon$ based on the definitions of SNR as follows:

$$\begin{aligned}
\mathrm{Cov}_t(d_t, v_t + \epsilon) &= \mathbf{E}_t[d_t]\mathbf{E}_t[v_t + \epsilon] \cdot \frac{\mathrm{Cov}_t(d_t, v_t + \epsilon)}{\sqrt{\mathrm{Var}_t(d_t)\mathrm{Var}_t(v_t + \epsilon)}} \sqrt{\frac{\mathrm{Var}_t(d_t)\mathrm{Var}_t(v_t + \epsilon)}{\mathbf{E}_t[d_t]^2 \mathbf{E}_t[v_t + \epsilon]^2}} \\
&= \mathbf{E}_t[d_t]\mathbf{E}_t[v_t + \epsilon] \cdot \frac{\mathrm{Corr}_t(d_t, v_t + \epsilon)}{\sqrt{\mathrm{SNR}_t(d_t)\mathrm{SNR}_t(v_t + \epsilon)}},
\end{aligned}$$

where $\mathrm{SNR}_t(d_t)$ is closely connected to the signal fraction ρ_t, defined as $\rho_t := \frac{\mathbf{E}_t[d_t]^2}{\mathbf{E}_t[d_t^2]}$:

$$\mathrm{SNR}_t(d_t) = \frac{\mathbf{E}_t[d_t]^2}{\mathrm{Var}_t(d_t)} = \frac{1}{\mathrm{Var}_t(d_t)/\mathbf{E}_t[d_t]^2} = \frac{1}{(\mathrm{Var}_t(d_t) + \mathbf{E}_t[d_t]^2)/\mathbf{E}_t[d_t]^2 - 1} = \frac{1}{1/\rho_t - 1}.$$

The first term (46) admits the following upper bound:

$$
\begin{aligned}
(46) &= \left|\frac{\mathbf{E}_t[d_t]}{\sqrt{\mathbf{E}_t[v_t]+\epsilon}}\right| \cdot \left|-\frac{\mathrm{Cov}_t(d_t, v_t+\epsilon)}{2\mathbf{E}_t[d_t]\mathbf{E}_t[v_t+\epsilon]} + \frac{3\mathrm{Var}_t(v_t+\epsilon)}{8\mathbf{E}_t[v_t+\epsilon]^2}\right| \\
&= \left|\frac{\mathbf{E}_t[d_t]}{\sqrt{\mathbf{E}_t[v_t]+\epsilon}}\right| \cdot \left|-\frac{\mathbf{E}[d_t]\mathbf{E}[v_t+\epsilon]}{2\mathbf{E}_t[d_t]\mathbf{E}_t[v_t+\epsilon]} \cdot \frac{\mathrm{Corr}_t(d_t, v_t+\epsilon)}{\sqrt{\mathrm{SNR}_t(v_t+\epsilon)}}\sqrt{\frac{1}{\rho_t}-1} + \frac{3}{8\mathrm{SNR}_t(v_t+\epsilon)}\right| \\
&= \left|\frac{\mathbf{E}_t[d_t]}{\sqrt{\mathbf{E}_t[v_t]+\epsilon}}\right| \cdot \left|-\frac{\mathrm{Corr}_t(d_t, v_t+\epsilon)}{2\sqrt{\mathrm{SNR}_t(v_t+\epsilon)}}\sqrt{\frac{1}{\rho_t}-1} + \frac{3}{8\mathrm{SNR}_t(v_t+\epsilon)}\right| \\
&\leq \frac{8+4\tau+3\tau^2}{8}\left|-\frac{\mathrm{Corr}_t(d_t, v_t+\epsilon)}{2\sqrt{\mathrm{SNR}_t(v_t+\epsilon)}}\sqrt{\frac{1}{\rho_t}-1} + \frac{3}{8\mathrm{SNR}_t(v_t+\epsilon)}\right| \cdot \left|\frac{\mathbf{E}_t[d_t]}{\sqrt{\mathbf{E}_t[d_t^2]}}\right| + \mathcal{O}(\epsilon)
\end{aligned} \tag{47}
$$

where (47) is given by Lemma D.1. □

Combining the upper bounds on two terms on the right-hand side of (41), we finally can prove Lemma 4.2

Proof of Lemma 4.2. It follows immediately by triangle inequality, Lemma D.1 and Lemma D.4.

$$
\begin{aligned}
&\left|\frac{\mathbf{E}_t[d_t]}{\sqrt{\mathbf{E}_t[d_t^2]}} - \mathbf{E}_t\left[\frac{d_t}{\sqrt{v_t+\epsilon}}\right]\right| \\
&\leq \left|\frac{\mathbf{E}_t[d_t]}{\sqrt{\mathbf{E}_t[d_t^2]}} - \frac{\mathbf{E}_t[d_t]}{\sqrt{\mathbf{E}_t[v_t]+\epsilon}}\right| + \left|\frac{\mathbf{E}_t[d_t]}{\sqrt{\mathbf{E}_t[v_t]+\epsilon}}\mathbf{E}_t\left[\frac{d_t}{\sqrt{v_t+\epsilon}}\right]\right| \\
&\leq \frac{4\tau+3\tau^2}{8}\left|\frac{\mathbf{E}_t[d_t]}{\sqrt{\mathbf{E}_t[d_t^2]}}\right| + \frac{8+4\tau+3\tau^2}{8}\left|\frac{\mathrm{Corr}_t(d_t, v_t+\epsilon)}{2\sqrt{\mathrm{SNR}_t(v_t+\epsilon)}}\sqrt{\frac{1}{\rho_t}-1} - \frac{3}{8\mathrm{SNR}_t[v_t+\epsilon]}\right| \cdot \left|\frac{\mathbf{E}_t[d_t]}{\sqrt{\mathbf{E}_t[d_t^2]}}\right| \\
&+ \mathcal{O}(\epsilon) + \mathcal{O}\left(\mathbf{E}_t[(Y-\mathbf{E}_t[Y])(Z-\mathbf{E}_t[Z])^2]\right) + \mathcal{O}\left(\mathbf{E}_t[(Z-\mathbf{E}_t[Z])^3]\right)
\end{aligned}
$$

Finally, bounding $|\mathrm{Corr}_t(d_t, v_t+\epsilon)|$ by 1 and recognizing $\frac{1}{\rho_t} - 1 = \frac{1}{\mathrm{SNR}_t(d_t)}$ give the desired result. □

D.2 Proof of Theorem 4.3

To prove Theorem 4.3, we can use a classical result on stochastic approximation originally due to Dvoretzky [15].

Theorem D.1 (An extension of Dvoretzky's Theorem)**.** *Let $(\Omega = \{\omega\}, \mathcal{F}, P)$ be a probability space. Let $\{x_t\}$ and $\{y_t\}$ be sequences of random variables such that, for all $t \geq 0$,*

$$
x_{t+1}(\omega) = T_t\big(x_0(\omega), \ldots, x_t(\omega)\big) + y_t(\omega), \tag{48}
$$

where the transformation T_t satisfy, for any $x_0, \ldots, x_t \in \mathbf{R}^n$,

$$
\big\|T_t(x_0, \ldots, x_t) - x_*\|^2 \leq \max\big\{a_t,\, (1+b_t)\|x_t - x_*\|^2 - c_t + d_t\big\} \tag{49}
$$

and the sequences $\{a_t\}$, $\{b_t\}$, $\{c_t\}$ and $\{d_t\}$ are non-negative and satisfy

$$
\lim_{t\to\infty} a_t = a_\infty, \qquad \sum_{t=0}^{\infty} b_t < \infty, \qquad \sum_{t=0}^{\infty} c_t = \infty, \qquad \sum_{t=1}^{\infty} d_t < \infty. \tag{50}
$$

In addition, suppose the following conditions hold with probability one:

$$
\mathbf{E}[\|x_0\|^2] < \infty, \qquad \sum_{t=0}^{\infty} \mathbf{E}[\|y_t\|^2] < \infty, \qquad \mathbf{E}\big[y_t | x_0, \ldots, x_t\big] = 0 \quad \forall\, t \geq 0.
$$

Then we have with probability one,

$$
\limsup_{t\to\infty} \|x_t - x_*\|^2 \leq a_\infty.
$$

Remark. There are many extensions of Dvoretzky's original results [15]. Theorem D.1 is a minor variation of Venter [51, Theorem 1]. More concretely,

- Theorem 1 of Venter [51] has the sequence $\{a_t\}$ being a constant sequence, i.e., $a_t = a_\infty$ for all $t \geq 0$. The extension to a non-constant sequence $\{a_t\}$ is outlined in the original work of Dvoretzky [15] and admits a simple proof due to Derman and Sacks [12].
- Theorem 1 of Venter [51] does not include the sequence $\{d_t\}$. The extension with $\sum_{t=0}^{\infty} d_t < \infty$ is straightforward based on a simple argument of Dvoretzky [15].
- More generally, the sequences $\{a_t\}$, $\{b_t\}$, $\{c_t\}$, $\{d_t\}$ can be non-negative measurable functions of $x_0, \ldots, x_t$, and the conclusion of Theorem D.1 holds if a_∞ is an upper bound on $\limsup_{t\to\infty} a_t(x_0, \ldots, x_t)$ uniformly for all sequences $x_0, \ldots, x_t, \ldots$ [12, 39].

We also need the following lemma.

Lemma D.5. *Under Assumptions B, it holds that*

$$\mathbf{E}_t\left[\left\langle x_t - x_*, \frac{d_t}{\sqrt{v_t+\epsilon}}\right\rangle\right] \geq \left\langle x_t - x_*, \frac{\mathbf{E}_t[d_t]}{\sqrt{\mathbf{E}_t[d_t^2]}}\right\rangle - \|x_t - x_*\|\left(c_t\|\sqrt{\rho_t}\| + \mathcal{O}(\epsilon) + \mathcal{O}(\mathrm{Var}_t(v_t))\right),$$

where c_t is given by (28)*.*

Proof. Adding and subtracting the term $\frac{\mathbf{E}_t[d_t]}{\sqrt{\mathbf{E}_t[d_t^2]}}$ from the inner product, we obtain:

$$\begin{aligned}
&\mathbf{E}_t\left[\left\langle x_t - x_*, \frac{d_t}{\sqrt{v_t+\epsilon}}\right\rangle\right] \\
=&\mathbf{E}_t\left[\left\langle x_t - x_*, \frac{\mathbf{E}_t[d_t]}{\sqrt{\mathbf{E}_t[d_t^2]}}\right\rangle\right] + \mathbf{E}_t\left[\left\langle x_t - x_*, \frac{d_t}{\sqrt{v_t+\epsilon}} - \frac{\mathbf{E}_t[d_t]}{\sqrt{\mathbf{E}_t[d_t^2]}}\right\rangle\right] \\
\geq&\mathbf{E}_t\left[\left\langle x_t - x_*, \frac{\mathbf{E}_t[d_t]}{\sqrt{\mathbf{E}_t[d_t^2]}}\right\rangle\right] - \|x_t - x_*\| \cdot \left\|\mathbf{E}_t\left[\frac{d_t}{\sqrt{v_t+\epsilon}}\right] - \frac{\mathbf{E}_t[d_t]}{\sqrt{\mathbf{E}_t[d_t^2]}}\right\| \\
\geq&\left\langle x_t - x_*, \frac{\mathbf{E}_t[d_t]}{\sqrt{\mathbf{E}_t[d_t^2]}}\right\rangle - \|x_t - x_*\|\left(c_t\left\|\frac{\mathbf{E}_t[d_t]}{\sqrt{\mathbf{E}_t[d_t^2]}}\right\| + \mathcal{O}(\epsilon) + \mathcal{O}(\mathrm{Var}_t(v_t))\right), \qquad (51)
\end{aligned}$$

where the inequality (51) is due to Lemma 4.2. To finish the proof, we recall $\sqrt{\rho_t} = \frac{\mathbf{E}_t[d_t]}{\sqrt{\mathbf{E}_t[d_t^2]}}$. □

Proof of Theorem 4.3. We can write the practical BCOSW algorithm as

$$\begin{aligned}
x_{t+1} &= (1-\alpha_t\lambda)x_t - \alpha_t\frac{d_t}{\sqrt{v_t+\epsilon}} \\
&= (1-\alpha_t\lambda)x_t - \alpha_t\mathbf{E}_t\left[\frac{d_t}{\sqrt{v_t+\epsilon}}\right] + \alpha_t\left(\mathbf{E}_t\left[\frac{d_t}{\sqrt{v_t+\epsilon}}\right] - \frac{d_t}{\sqrt{v_t+\epsilon}}\right)
\end{aligned}$$

In terms of the decomposition in (48), we have $x_{t+1} = T_t(x_0, \ldots, x_t) + y_t$ where

$$\begin{aligned}
T_t(x_0, \ldots, x_t) &= (1-\alpha_t\lambda)x_t - \alpha_t\mathbf{E}_t\left[\frac{d_t}{\sqrt{v_t+\epsilon}}\right] \\
y_t &= \alpha_t\left(\mathbf{E}_t\left[\frac{d_t}{\sqrt{v_t+\epsilon}}\right] - \frac{d_t}{\sqrt{v_t+\epsilon}}\right).
\end{aligned}$$

Apparently we have $\mathbf{E}_t[y_t] = \mathbf{E}[y_t|x_0, \ldots, x_t] = 0$. We also have $\sum_{t=0}^{\infty}\mathbf{E}[\|y_t\|^2] < \infty$ with a bounded assumption on y_t due to the assumption $\sum_{t=0}^{\infty}\alpha_t^2 < \infty$.

The squared distance between $T_t(x_0, \ldots, x_t)$ and x_* is

$$\begin{aligned}\left\|T_t(x_0,\ldots,x_t) - x_*\right\|^2 &= \left\|(1-\alpha_t\lambda)x_t - \alpha_t \mathbf{E}_t\left[\frac{d_t}{\sqrt{v_t+\epsilon}}\right] - x_*\right\|^2 \\ &= (1-\alpha_t\lambda)^2\|x_t - x_*\|^2 - 2\alpha_t(1-\alpha_t\lambda)\left\langle x_t - x_*,\, \mathbf{E}_t\left[\frac{d_t}{\sqrt{v_t+\epsilon}}\right] + \lambda x_*\right\rangle \\ &\quad + \alpha_t^2\left\|\mathbf{E}_t\left[\frac{d_t}{\sqrt{v_t+\epsilon}}\right] + \lambda x_*\right\|^2\end{aligned}$$

From Lemma D.5 and the aiming condition (36), we have

$$\begin{aligned}\left\langle x_t - x_*,\, \mathbf{E}_t\left[\frac{d_t}{\sqrt{v_t+\epsilon}}\right] + \lambda x_*\right\rangle &\geq \left\langle x_t - x_*,\, \frac{\mathbf{E}_t[d_t]}{\sqrt{\mathbf{E}_t[d_t^2]}} + \lambda x_*\right\rangle - \Big(c_t\|\sqrt{\rho_t}\| + \mathcal{O}(\epsilon) + \mathcal{O}(\mathrm{Var}_t(v_t))\Big)\|x_t - x_*\| \\ &\geq -\Big(c_t\|\sqrt{\rho_t}\| + \mathcal{O}(\epsilon) + \mathcal{O}(\mathrm{Var}_t(v_t))\Big)\|x_t - x_*\|.\end{aligned}$$

In addition, by the bounded assumption on d_t, there exist a constant B such that

$$\left\|\mathbf{E}_t\left[\frac{d_t}{\sqrt{v_t+\epsilon}}\right] + \lambda x_*\right\|^2 \leq B, \qquad \forall\, t \geq 0.$$

Together with $0 < 1 - \alpha_t\lambda < 1$, we conclude that

$$\begin{aligned}\left\|T_t(x_0,\ldots,x_t) - x_*\right\|^2 &\leq (1-\alpha_t\lambda)^2\|x_t - x_*\|^2 + 2\alpha_t(1-\alpha_t\lambda)\Big(c_t\|\sqrt{\rho_t}\| + \mathcal{O}(\epsilon) + \mathcal{O}(\mathrm{Var}_t(v_t))\Big)\|x_t - x_*\| + \alpha_t^2 B \\ &\leq (1-\alpha_t\lambda)^2\|x_t - x_*\|^2 + 2\alpha_t\Big(c_t\|\sqrt{\rho_t}\| + \mathcal{O}(\epsilon) + \mathcal{O}(\mathrm{Var}_t(v_t))\Big)\|x_t - x_*\| + \alpha_t^2 B \\ &= (1+\alpha_t^2\lambda^2)\|x_t - x_*\|^2 + \alpha_t\big(2c\|\sqrt{\rho_t}\| + \mathcal{O}(\epsilon) + \mathcal{O}(\mathrm{Var}_t(v_t))\big)\|x_t - x_*\| \\ &\quad - 2\alpha_t\lambda\|x_t - x_*\|^2 + \alpha_t^2 B \\ &= (1+\alpha_t^2\lambda^2)\|x_t - x_*\|^2 + \alpha_t\Big(\big(2c\|\sqrt{\rho_t}\| + \mathcal{O}(\epsilon) + \mathcal{O}(\mathrm{Var}_t(v_t))\big)\|x_t - x_*\| - \lambda\|x_t - x_*\|^2\Big) \\ &\quad - \alpha_t\lambda\|x_t - x_*\|^2 + \alpha_t^2 B.\end{aligned}$$

We observe that there exist $\delta > 0$ such that

$$\|x_t - x_*\| \geq \delta \quad \Longrightarrow \quad \big(2c\|\sqrt{\rho_t}\| + \mathcal{O}(\epsilon) + \mathcal{O}(\mathrm{Var}_t(v_t))\big)\|x_t - x_*\| - \lambda\|x_t - x_*\|^2 \leq 0.$$

Therefore, $\|x_t - x_*\| \geq \delta$ implies

$$\|T_t(x_0,\ldots,x_t) - x_*\|^2 \leq (1+\alpha_t^2\lambda^2)\|x_t - x_*\|^2 - \alpha_t\lambda\|x_t - x_*\|^2 + \alpha_t^2 B.$$

Otherwise, when $\|x_t - x_*\| \leq \delta$, we have

$$\|T_t(x_0,\ldots,x_t) - x_*\|^2 \leq (1-\alpha_t\lambda)^2\delta^2 + 2\alpha_t\big(c\|\sqrt{\rho_t}\|\delta + \mathcal{O}(\epsilon)\big) + \alpha_t^2 B.$$

By defining a_t as the right-hand side of the above inequality, i.e.,

$$a_t = (1-\alpha_t\lambda)^2\delta^2 + 2\alpha_t\big(c\|\sqrt{\rho_t}\|\delta + \mathcal{O}(\epsilon)\big) + \alpha_t^2 B, \tag{52}$$

we can combine the above two cases as

$$\|T_t(x_0,\ldots,x_t) - x_*\|^2 \leq \max\big\{a_t,\, (1+\alpha_t^2\lambda^2)\|x_t - x_*\|^2 - \alpha_t\lambda\|x_t - x_*\|^2 + \alpha_t^2 B\big\}.$$

With the additional definition of

$$b_t = \alpha_t^2\lambda^2, \qquad c_t = \alpha_t\lambda\|x_t - x_*\|^2, \qquad d_t = \alpha_t^2 B,$$

we arrive at the key inequality (49).

We are left to check the conditions in (50). Using the assumptions on $\{\alpha_t\}$, the definition in (52) implies that a_t converges and $\lim_{t\to\infty} a_t = \delta^2$. The conditions on $\{b_t\}$ and $\{d_t\}$ are automatically satisfied. For $\{c_t\}$, if $\sum_{t=0}^{\infty} c_t < \infty$, then we must have $\|x_t - x_\star\|^2 \to 0$ almost surely and the conclusion of theorem holds trivially. Otherwise, $\sum_{t=0}^{\infty} c_t = \infty$ allows all the conditions in (50) to hold, so we can invole Theorem D.1 to conclude the proof. □

E Biases and variances of second-moment estimators

Lemma 4.2 provides guidelines for choosing the second-moment estimator v_t, which should exhibit:

- low bias (i.e., low τ);
- high signal-to-noise ratio (i.e., high SNR);

However, there is always a bias-variance tradeoff for various estimators v_t. Here are some examples:

1. **Sign-SGD** is equivalent to take $v_t = d_t^2$, exhibiting low bias and high variance

$$\text{Bias} = |\mathbf{E}_t[v_t] - \mathbf{E}_t[d_t^2]| = 0, \qquad \text{Variance} = \text{Var}_t(v_t) = \text{Var}_t(d_t),$$

   with resulting update rule to be sign-SGD (with and without momentum corresponding to $d_t = g_t$ and $d_t = m_t$ respectively):

$$x_{t+1} = x_t - \alpha_t \frac{d_t}{\sqrt{d_t^2 + \epsilon}} \approx x_t - \alpha_t \text{sign}(d_t).$$

2. **Standard SGD** is equivalent to take $v_t = c$ for some positive constant c, exhibiting high bias and low variance

$$\text{Bias} = |\mathbf{E}_t[v_t] - \mathbf{E}_t[d_t^2]| = |c - \mathbf{E}_t[d_t^2]|, \qquad \text{Variance} = \text{Var}_t(v_t) = 0,$$

   with resulting update rule to be SGD (with and without momentum corresponding to $d_t = g_t$ and $d_t = m_t$ respectively):

$$x_{t+1} = x_t - \alpha_t \frac{d_t}{\sqrt{c + \epsilon}} = x_t - \alpha_t' d_t,$$

   where $\alpha_t' := \frac{\alpha_t}{\sqrt{c+\epsilon}}$.

3. **BCOS-m** uses $v_t = \text{EMA}_\beta(d_t^2)$, exhibiting non-trivial bias and low variance properties:

$$\begin{aligned}
\text{Bias} &= |\mathbf{E}_t[v_t] - \mathbf{E}_t[d_t^2]| \\
&= \left| \mathbf{E}_t \left[\sum_{k=1}^{t} (1-\beta)\beta^{t-k} d_k^2 \right] - \mathbf{E}_t[d_t^2] \right| \\
&= \left| \sum_{k=1}^{t-1} (1-\beta)\beta^{t-k} d_k^2 + (1-\beta)\mathbf{E}_t\left[d_t^2\right] - \mathbf{E}_t[d_t^2] \right| \\
&= \left| \sum_{k=1}^{t-1} (1-\beta)\beta^{t-k} d_k^2 - \beta \mathbf{E}_t\left[d_t^2\right] \right|.
\end{aligned}$$

   As for the variance, we get

$$\begin{aligned}
\text{Var}_t(v_t) &= \mathbf{E}_t \left[\left(\sum_{k=1}^{t} (1-\beta)\beta^{t-k} d_k^2 - \sum_{k=1}^{t} (1-\beta)\beta^{t-k} \mathbf{E}_t[d_k^2] \right)^2 \right] \\
&= \mathbf{E}_t \left[\left(\sum_{k=1}^{t-1} (1-\beta)\beta^{t-k} d_k^2 + (1-\beta) d_t^2 - \sum_{k=1}^{t-1} (1-\beta)\beta^{t-k} d_k^2 - (1-\beta)\mathbf{E}_t[d_k^2] \right)^2 \right] \\
&= (1-\beta)^2 \mathbf{E}_t \left[\left(d_t^2 - \mathbf{E}_t[d_t^2] \right)^2 \right] \\
&= (1-\beta)^2 \text{Var}_t\left(d_t^2\right).
\end{aligned}$$

4. **Adam** uses estimator $v_t = \text{EMA}_{\beta_2}(g_t^2)$ with search direction $d_t = \text{EMA}_{\beta_1}(g_t)$: exhibiting non-trivial bias and low variance properties:

$$\text{Bias} = |\mathbf{E}_t[v_t] - \mathbf{E}_t[d_t^2]|$$

$$
\begin{aligned}
&= \left| \mathbf{E}_t \left[\sum_{k=1}^{t} (1-\beta_2)\beta_2^{t-k} g_k^2 \right] - \mathbf{E}_t \left[\left(\sum_{k=1}^{t} (1-\beta_1)\beta_1^{t-k} g_k \right)^2 \right] \right| \\
&\leq \left| \sum_{k=1}^{t-1} (1-\beta_2)\beta_2^{t-k} g_k^2 - \left(\sum_{k=1}^{t-1} (1-\beta_1)\beta_1^{t-k} g_k \right)^2 \right| \\
&\quad + 2 \left| \left(\sum_{k=1}^{t-1} (1-\beta_1)\beta_1^{t-k} g_k \right) \mathbf{E}_t[g_t] \right| + \left| (1-\beta_2)\mathbf{E}_t\left[g_t^2\right] - (1-\beta_1)^2 \mathbf{E}_t[g_t^2] \right|.
\end{aligned}
$$

As for the variance, we get

$$
\begin{aligned}
\text{Var}_t(v_t) &= \mathbf{E}_t \left[\left(\sum_{k=1}^{t} (1-\beta_2)\beta_2^{t-k} g_k^2 - \sum_{k=1}^{t} (1-\beta_2)\beta_2^{t-k} \mathbf{E}_t\left[g_k^2\right] \right)^2 \right] \\
&= \mathbf{E}_t \left[\left(\sum_{k=1}^{t-1} (1-\beta_2)\beta_2^{t-k} g_k^2 + (1-\beta_2) g_t^2 - \sum_{k=1}^{t-1} (1-\beta_2)\beta_2^{t-k} g_k^2 - (1-\beta_2)\mathbf{E}_t[g_t^2] \right)^2 \right] \\
&= (1-\beta_2)^2 \mathbf{E}_t \left[\left(d_t^2 - \mathbf{E}_t[d_t^2] \right)^2 \right] \\
&= (1-\beta_2)^2 \text{Var}_t\left(d_t^2 \right).
\end{aligned}
$$

5. **BCOS-c** uses estimator $v_t = (1-(1-\beta)^2)m_{t-1}^2 + (1-\beta)^2 g_t^2$ with search direction $d_t = m_t = \text{EMA}_\beta(g_t) = \beta m_{t-1} + (1-\beta)g_t$: exhibiting low bias and low variance properties:

$$
\begin{aligned}
\text{Bias} &= |\mathbf{E}_t[v_t] - \mathbf{E}_t[d_t^2]| \\
&= \left| \mathbf{E}_t \left[(1-(1-\beta)^2)m_{t-1}^2 + (1-\beta)^2 g_t^2 \right] - \mathbf{E}_t \left[\left(\beta m_{t-1} + (1-\beta)g_t \right)^2 \right] \right| \\
&= \left| (2\beta - \beta^2)m_{t-1}^2 + (1-\beta)^2 \mathbf{E}_t\left[g_t^2\right] - \beta^2 m_{t-1}^2 - 2\beta(1-\beta)m_{t-1}\mathbf{E}_t\left[g_t\right] - (1-\beta)\mathbf{E}_t\left[g_t^2\right] \right| \\
&= \left| (2\beta - 2\beta^2)m_{t-1}^2 - 2\beta(1-\beta)m_{t-1}\mathbf{E}_t\left[g_t\right] \right| \\
&= 2\beta(1-\beta)\left| m_{t-1}\left(m_{t-1} - \mathbf{E}_t\left[g_t\right] \right) \right|
\end{aligned}
$$

$$
\begin{aligned}
\text{Var}_t(v_t) &= \mathbf{E}_t \left[\left((1-(1-\beta)^2)m_{t-1}^2 + (1-\beta)^2 g_t^2 - (1-(1-\beta)^2)m_{t-1}^2 - (1-\beta)^2 \mathbf{E}_t[g_t^2] \right)^2 \right] \\
&= (1-\beta)^4 \mathbf{E}_t \left[\left(g_t^2 - \mathbf{E}_t[g_t^2] \right)^2 \right] \\
&= (1-\beta)^4 \text{Var}_t\left(g_t^2 \right).
\end{aligned}
$$

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: *The claims match theoretical and experimental results presented.*

   Guidelines:

   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: *We discuss the need of tuning the stepsize schedule in Sections 2.2, 2.3 and 5.*

   Guidelines:

   - The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [Yes]

   Justification: *We state the assumptions and key results clearly and give rigorous proofs.*

Guidelines:

- The answer NA means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: *We describe the hyper-parameters used in the experiments clearly, and included an optimizer implementation in Appendix A that is used to generate the experiment results.*

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
     (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [Yes]

   Justification: *The data used in this paper are all widely available in the public domain. We also include the optimizer code in Appendix A for reproducibility.*

Guidelines:

- The answer NA means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so "No" is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

Answer: [Yes]

Justification: *We describe the hyper-parameters used in the Experiment section. The models and datasets we use are all very standard and there should be no confusion on the settings.*

Guidelines:

- The answer NA means that the paper does not include experiments.
- The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
- The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

Answer: [No]

Justification: *Performing the errors bars are computationally costly and they are not essential in understanding and justifying the results in this paper.*

Guidelines:

- The answer NA means that the paper does not include experiments.
- The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.

- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [No]

   Justification: *Information on compute resources are not relevant to the results of this paper. We focus on training performance of standard tasks whose compute requirements are well-known.*

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: *We reviewed the Code of Ethics and stick with it.*

   Guidelines:

   - The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [NA]

    Justification: *There is no perceivable negative impact of the work perfomed.*

    Guidelines:

    - The answer NA means that there is no societal impact of the work performed.
    - If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
    - The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.

- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

    Answer: [NA]

    Justification: *This paper poses no such risks.*

    Guidelines:

    - The answer NA means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: *All sources are properly cited and no license required.*

    Guidelines:

    - The answer NA means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
    - For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
    - If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [NA]

    Justification: *This paper does not release new assets.*

    Guidelines:

    - The answer NA means that the paper does not release new assets.

- Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
- The paper should discuss whether and how consent was obtained from people whose asset is used.
- At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [NA]

    Justification: *This paper does not involve crowdsourcing nor research with human subjects.*

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [NA]

    Justification: *This paper does not involve crowdsourcing nor research with human subjects.*

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
    - For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does not impact the core methodology, scientific rigorousness, or originality of the research, declaration is not required.

    Answer: [NA]

    Justification: *the core method development in this research does not involve LLMs as any important, original, or non-standard components.*

    Guidelines:

    - The answer NA means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy (`https://neurips.cc/Conferences/2025/LLM`) for what should or should not be described.