ON THE PERFORMANCE OF RANDOM RESHUFFLING IN STOCHASTIC LEARNING

Bicheng Ying *Kun Yuan* *Stefan Vlaski* *Ali H. Sayed*

Department of Electrical Engineering
University of California, Los Angeles

ABSTRACT

In empirical risk optimization, it has been observed that gradient descent implementations that rely on random reshuffling of the data achieve better performance than implementations that rely on sampling the data randomly and independently of each other. Recent works have pursued justifications for this behavior by examining the convergence rate of the learning process under diminishing step-sizes. Some of these justifications rely on loose bounds, or their conclusions are dependent on the sample size which is problematic for large datasets. This work focuses on constant step-size adaptation, where the agent is continuously learning. In this case, convergence is only guaranteed to a small neighborhood of the optimizer albeit at a linear rate. The analysis establishes analytically that random reshuffling outperforms independent sampling by showing that the iterate at the end of each run approaches a smaller neighborhood of size $O(\mu^2)$ around the minimizer rather than $O(\mu)$. Simulation results illustrate the theoretical findings.

Index Terms— Random reshuffling, stochastic gradient descent, mean-square performance, convergence analysis.

1. THE RANDOM RESHUFFLING IMPLEMENTATION

We consider minimizing an empirical risk function $J(w)$, which is defined as a sample average of loss values over a possibly large but finite training set:

$$w^\star \triangleq \arg\min_{w\in\mathbb{R}^M} \quad J(w) \triangleq \frac{1}{N}\sum_{n=1}^{N} Q(w;x_n), \tag{1}$$

where the $\{x_n\}_{n=1}^N$ are training data samples and the loss functions $Q(w;x_n)$ are assumed differentiable. We assume the empirical risk $J(w)$ is strongly-convex so that the minimizer, $w^\star$, is unique. Problems of the form (1) are common in many areas of machine learning including linear regression, logistic regression and their regularized versions.

When the size of the dataset N is large, it is impractical to solve (1) directly with traditional gradient descent. One simple, yet powerful approach is to employ the stochastic gradient method (SGD) [1–7]. In this method, at every iteration, rather than compute the full gradient $\nabla_w J(w)$ on the entire data set, the algorithm picks one index $\boldsymbol{n}_i$ at random, and employs $\nabla_w Q(w;x_{\boldsymbol{n}_i})$ to approximate $\nabla_w J(w)$. Specifically, at iteration i, the update for estimating the minimizer is of the form [8]:

$$\boldsymbol{w}_i = \boldsymbol{w}_{i-1} - \mu\nabla_w Q(\boldsymbol{w}_{i-1};x_{\boldsymbol{n}_i}), \tag{2}$$

This work was supported in part by NSF grants CCF-1524250 and ECCS-1407712. Emails:{ybc,kunyuan,svlaski,sayed}@ucla.edu

where μ is the step-size parameter. Note that we are using boldface notation to refer to random variables. Normally, the index $\boldsymbol{n}_i$ is uniformly distributed over the discrete set $\{1,2,\ldots,N\}$.

However, it has been noted in the literature [9–12] that incorporating random reshuffling into the gradient descent implementation helps achieve better performance. In a random reshuffling implementation, the gradient descent algorithm is run multiple times over the data where each run is indexed by $k \geq 1$ and is referred to as an epoch. For each run, the original data is first reshuffled so that the sample of index i becomes the sample of index $\boldsymbol{\sigma}^k(i)$, where the symbol $\boldsymbol{\sigma}$ represents a uniform random permutation of the indices. In this way, we can express the random reshuffling algorithm for the $k-$th run in the following manner:

$$\boldsymbol{w}_i^k = \boldsymbol{w}_{i-1}^k - \mu\nabla_w Q(\boldsymbol{w}_{i-1}^k;x_{\boldsymbol{\sigma}^k(i)}), \quad i=1,\ldots,N \tag{3}$$

with the boundary condition:

$$\boldsymbol{w}_0^k = \boldsymbol{w}_N^{k-1} \tag{4}$$

In other words, the initial condition for run k is the last iterate from run $k-1$. The boldface notation for the symbols $\boldsymbol{w}$ and $\boldsymbol{\sigma}$ in (3) is meant to emphasize the random nature of these variables due to the randomness in the permutation operation. The uniformity of the permutation function implies the following useful properties:

$$\boldsymbol{\sigma}^k(i) \neq \boldsymbol{\sigma}^k(j), \quad 1\leq i\neq j\leq N \tag{5}$$

$$\mathbb{P}[\,\boldsymbol{\sigma}^k(i)=n\,] = \frac{1}{N}, \qquad 1\leq n\leq N \tag{6}$$

$$\mathbb{P}[\boldsymbol{\sigma}^k(i+1)=n\,|\,\boldsymbol{\sigma}^k(1\colon i)] = \begin{cases} \dfrac{1}{N-i}, & n\notin\boldsymbol{\sigma}^k(1{:}i) \\ 0 \quad , & n\in\boldsymbol{\sigma}^k(1{:}i) \end{cases} \tag{7}$$

where $\boldsymbol{\sigma}^k(1{:}i)$ represents the collection of permuted indices for the original samples numbered 1 through i.

Recent works [10, 11, 13] have pursued justifications for the enhanced behavior of random reshuffling implementations over independent sampling (with replacement) by examining the convergence rate of the learning process under diminishing step-sizes. Some of these justifications rely on loose bounds, or their conclusions are dependent on the sample size which is problematic for large datasets. Also, some of the results only establish that random reshuffling will not degrade performance relative to the stochastic gradient descent implementation. In this work, we focus on constant step-size adaptation, where the agent is continuously learning. In this case, convergence is only guaranteed to a small neighborhood of the optimizer albeit at a linear rate. The analysis will establish analytically that random reshuffling outperforms independent sampling (with replacement) by showing that the mean-square-error of the iterate at

the end of each run in the random reshuffling strategy will be in the order of $O(\mu^2)$ rather than $O(\mu)$, which is a significant improvement. Simulation results will illustrate this conclusion.

1.1. Weight-Error Dynamics

To analyze the behavior of the reshuffling algorithm (3), we first introduce the gradient noise process, which is the difference between the true gradient of the empirical risk and its approximation by the gradient of the loss function, i.e., we rewrite (3) in the form:

$$\begin{aligned}\boldsymbol{w}_i^k = \; & \boldsymbol{w}_{i-1}^k - \mu\nabla_w J(\boldsymbol{w}_{i-1}^k) + \\ & \mu\underbrace{\Big[\nabla_w J(\boldsymbol{w}_{i-1}^k) - \nabla_w Q(\boldsymbol{w}_{i-1}^k; x_{\boldsymbol{\sigma}^k(i)})\Big]}_{\triangleq\, s_{\boldsymbol{\sigma}^k(i)}(\boldsymbol{w}_{i-1}^k)}\end{aligned} \tag{8}$$

where the notation $s_{\boldsymbol{\sigma}^k(i)}(\cdot)$ refers to the gradient noise process. One main difficulty for the analysis in the subsequent derivations arises from the fact that the gradient noise $s_{\boldsymbol{\sigma}^k(i)}(\boldsymbol{w}_{i-1}^k)$ is not independent of past selections, $\boldsymbol{\sigma}^k(1{:}i-1)$. However, this same noise is independent of index choices over different epochs, $\boldsymbol{\sigma}^{k'}(1{:}i-1)$, for $k' \neq k$. For ease of reference, we introduce the error vector and the Hessian matrix of the empirical risk at the optimizer and denote them by:

$$\widetilde{\boldsymbol{w}}_i^k \triangleq w^\star - \boldsymbol{w}_i^k \tag{9}$$

$$H \triangleq \nabla_w^2 J(w^\star) \tag{10}$$

Assumption 1 (CONDITION ON LOSS FUNCTION)**.** *It is assumed that $Q(w;x_n)$ is differentiable and has a δ_n-Lipschitz continuous gradient, i.e., for every $n=1,\ldots,N$ and any $w_1, w_2 \in \mathbb{R}^M$:*

$$\|\nabla_w Q(w_1;x_n) - \nabla_w Q(w_2;x_n)\| \leq \delta_n \|w_1 - w_2\| \tag{11}$$

where $\delta_n > 0$. We also assume $J(w)$ is ν-strongly convex:

$$\Big(\nabla_w J(w_1) - \nabla_w J(w_2)\Big)^{\mathsf{T}}(w_1 - w_2) \geq \nu\|w_1 - w_2\|^2 \tag{12}$$

■

If we introduce $\delta = \max\{\delta_1, \delta_2, \cdots, \delta_N\}$, then each $\nabla_w Q(w;x_n)$ and $\nabla_w J(w)$ are also δ-Lipschitz continuous.

Assumption 2 (HESSIAN IS LIPSCHITZ CONTINUOUS)**.** *The risk function $J(w)$ has a Lipschitz continuous Hessian matrix, i.e., there exists a constant $\kappa \geq 0$, such that*

$$\|\nabla_w^2 J(w_1) - \nabla_w^2 J(w_2)\| \leq \kappa\|w_1 - w_2\| \tag{13}$$

■

Under this last assumption, the gradient vector, $\nabla_w J(w)$, can be expressed in Taylor expansion in the form [14, p. 378]:

$$\nabla_w J(w) = \nabla_w^2 J(w^\star)(w - w^\star) + \xi(w), \quad \forall w \tag{14}$$

where the residue term satisfies:

$$\|\xi(w)\| \leq \frac{\kappa}{2}\|w - w^\star\|^2 \tag{15}$$

Subtracting $w^\star$ from both sides of (8) gives

$$\widetilde{\boldsymbol{w}}_i^k = (I - \mu H)\widetilde{\boldsymbol{w}}_{i-1}^k - \mu s_{\boldsymbol{\sigma}^k(i)}(\boldsymbol{w}_{i-1}^k) + \mu\xi(\boldsymbol{w}_{i-1}^k) \tag{16}$$

1.2. Properties of the Gradient Noise Process

Recursion (16) describes the evolution of the error dynamics of the learning algorithm. To proceed with the analysis, we need to highlight some properties of the gradient noise process.

To begin with, we observe that, conditioned on prior data, the gradient noise is generally biased since

$$\begin{aligned}&\mathbb{E}\left[s_{\boldsymbol{\sigma}^k(i)}(\boldsymbol{w}_{i-1}^k) \,|\, \boldsymbol{w}_{i-1}^k, \boldsymbol{\sigma}^k(1{:}i-1)\right] \\ &= \frac{1}{N-i+1}\sum_{n\notin\boldsymbol{\sigma}^k(1\,:\,i-1)} s_n(\boldsymbol{w}_{i-1}^k) \\ &= \nabla_w J(\boldsymbol{w}_{i-1}^k) - \frac{1}{N-i+1}\sum_{n\notin\boldsymbol{\sigma}^k(1\,:\,i-1)} Q(\boldsymbol{w}_{i-1}^k; x_n)\end{aligned} \tag{17}$$

and the difference (17) is nonzero in general in view of the definition (1). In comparison, it is easy to check that the following conditional mean is zero:

$$\begin{aligned}\mathbb{E}\left[s_{\boldsymbol{\sigma}^k(i)}(\boldsymbol{w}_0^k)\,\Big|\,\boldsymbol{w}_0^k\right] &\overset{(6)}{=} \frac{1}{N}\sum_{n=1}^{N}[\nabla J(\boldsymbol{w}_0^k) - Q(\boldsymbol{w}_0^k; x_n)] \\ &= 0\end{aligned} \tag{18}$$

This second property motivates us to expand (16) into the following error recursion by adding and subtracting the same gradient noise term evaluated at $\boldsymbol{w}_0^k$:

$$\begin{aligned}\widetilde{\boldsymbol{w}}_i^k =& (I - \mu H)\widetilde{\boldsymbol{w}}_{i-1}^k - \mu s_{\boldsymbol{\sigma}^k(i)}(\boldsymbol{w}_0^k) \\ & - \underbrace{\mu\big(s_{\boldsymbol{\sigma}^k(i)}(\boldsymbol{w}_{i-1}^k) - s_{\boldsymbol{\sigma}^k(i)}(\boldsymbol{w}_0^k)\big)}_{\text{noise mismatch}} + \mu\xi(\boldsymbol{w}_{i-1}^k)\end{aligned} \tag{19}$$

Iterating (19) and using (4) we can establish the following useful relation, which we call upon in the sequel:

$$\begin{aligned}\widetilde{\boldsymbol{w}}_0^{k+1} = \; & (I - \mu H)^N \widetilde{\boldsymbol{w}}_0^k - \mu\sum_{i=1}^{N}(I - \mu H)^{N-i} s_{\boldsymbol{\sigma}^k(i)}(\boldsymbol{w}_0^k) \\ & - \mu\sum_{i=1}^{N}(I - \mu H)^{N-i}\Big(s_{\boldsymbol{\sigma}^k(i)}(\boldsymbol{w}_{i-1}^k) - s_{\boldsymbol{\sigma}^k(i)}(\boldsymbol{w}_0^k)\Big) \\ & + \mu\sum_{i=1}^{N}(I - \mu H)^{N-i}\xi(\boldsymbol{w}_{i-1}^k)\end{aligned} \tag{20}$$

2. CONVERGENCE ANALYSIS

We next provide two results that establish the stability and performance of the random reshuffling algorithm. The first lemma below establishes in (21) the convergence of every iterate $\boldsymbol{w}_i^k$ to a neighborhhod of size $O(\mu)$ around $w^\star$ for inifinitely many epoch runs. The second lemma focuses on the convergence of the *starting* point of each epoch and establishes in (31) that it actually approaches a smaller neighborhood of size $O(\mu^2)$ around $w^\star$.

Lemma 1 (ACCURACY OF ITERATES)**.** *Under assumptions 1 and 2, it holds for sufficiently small step-sizes that*

$$\limsup_{k\to\infty} \mathbb{E}\,\|\boldsymbol{w}_i^k - w^\star\|^2 = O(\mu), \quad 1 \leq i \leq N \tag{21}$$

$$\limsup_{k\to\infty} \mathbb{E}\,\|\boldsymbol{w}_i^k - w^\star\|^4 = O(\mu^2), \quad 1 \leq i \leq N \tag{22}$$

Proof. In a manner similar to (18), we can verify that

$$\mathbb{E}_{\boldsymbol{\sigma}}[\boldsymbol{s}_{\boldsymbol{\sigma}^k(i)}(\boldsymbol{w}_{i-1}^k) \,|\, \boldsymbol{w}_{i-1}^k] = 0, \quad \forall\, i, k \tag{23}$$

Now, using the mean-value theorem [14, p.744], we can rewrite (8) into:

$$\widetilde{\boldsymbol{w}}_i^k = (I - \mu \boldsymbol{H}_{i-1}^k)\widetilde{\boldsymbol{w}}_{i-1}^k - \mu s_{\boldsymbol{\sigma}^k(i)}(\boldsymbol{w}_{i-1}^k) \tag{24}$$

where

$$\boldsymbol{H}_{i-1}^k \triangleq \int_0^1 \nabla_w^2 J(w^\star - r\widetilde{\boldsymbol{w}}_{i-1}^k)dr \tag{25}$$

After squarring (24), taking the expectation conditioned on $\boldsymbol{w}_{i-1}^k$, and cancelling out the cross-term using (23), we obtain:

$$\begin{aligned}\mathbb{E}\,[\|\widetilde{\boldsymbol{w}}_i^k\|^2 \,|\, \boldsymbol{w}_{i-1}^k] \leq \|I - \mu \boldsymbol{H}_{i-1}^k\|^2 \|\widetilde{\boldsymbol{w}}_{i-1}^k\|^2 + \\ \mu^2 \mathbb{E}\,\|s_{\boldsymbol{\sigma}^k(i)}(\boldsymbol{w}_{i-1}^k) \,|\, \boldsymbol{w}_{i-1}^k\|^2\end{aligned} \tag{26}$$

Note that it is critical to condition only on $\boldsymbol{w}_{i-1}^k$, and not on $\boldsymbol{\sigma}^k(1{:}i-1)$, in order to remove the cross-term. Otherwise, the cross-term will not be zero because of (17). Next, we recall that

$$\|I - \mu \boldsymbol{H}_{i-1}^k\|^2 \leq \max\{(1-\mu\delta)^2, (1-\mu\nu)^2\} = 1 - O(\mu) \tag{27}$$

where ν and δ are the strongly-convex and gradient Lipschitz constants for the risk function $J(w)$. Moreover, the gradient noise variance satisfies:

$$\begin{aligned}\mathbb{E}\,\|s_{\boldsymbol{\sigma}^k(i)}(\boldsymbol{w}_{i-1}^k) \,|\, \boldsymbol{w}_{i-1}^k\|^2 &= \frac{1}{N}\sum_{n=1}^N \|s_n(\boldsymbol{w}_{i-1}^k)\|^2 \\ &\leq \beta_e^2 \|\widetilde{\boldsymbol{w}}_{i-1}^k\|^2 + \sigma_e^2\end{aligned} \tag{28}$$

where β_e^2 and σ_e^2 are some data-related non-negative constants and the inequality in (28) was established in [8, Lemma 1]. Combining (24), (27), and (28), we conclude that

$$\limsup_{k\to\infty} \mathbb{E}\,\|\widetilde{\boldsymbol{w}}_i^k\|^2 = O(\mu) \tag{29}$$

Likewise, using an argument similar to [14, pp. 352-355] we can establish the validity of (22). □

We can provide a more accurate bound about the size of the error for the *starting* points of the various runs by exploiting another useful property of the gradient noise process, namely, the fact that

$$\frac{1}{N}\sum_{i=1}^N s_{\boldsymbol{\sigma}^k(i)}(w) = \frac{1}{N}\sum_{i=1}^N s_i(w) \equiv 0, \qquad \forall w \tag{30}$$

This property does not hold for traditional stochastic gradient descent implementations with data replacement.

Lemma 2 (ACCURACY OF STARTING POINTS)**.** *Under assumptions 1 and 2, the starting point of each run satisfies*

$$\limsup_{k\to\infty} \mathbb{E}\,\|\boldsymbol{w}_0^k - w^\star\|^2 = O(\mu^2) \tag{31}$$

Proof. Squaring (20), conditioning on $\widetilde{\boldsymbol{w}}_0^k$, and using Jensen's inequality gives:

$$\begin{aligned}&\mathbb{E}\left[\|\widetilde{\boldsymbol{w}}_0^{k+1}\|^2 \,|\, \widetilde{\boldsymbol{w}}_0^k\right] \\ &\leq \frac{1}{t}\underbrace{\mathbb{E}\left\|(I-\mu H)^N \widetilde{\boldsymbol{w}}_0^k - \mu\sum_{i=1}^N (I-\mu H)^{N-i} s_{\boldsymbol{\sigma}^k(i)}(\boldsymbol{w}_0^k)\right\|^2}_{\triangleq A} \\ &\quad + \frac{2\mu^2}{1-t}\underbrace{\mathbb{E}\left\|\sum_{i=1}^N (I-\mu H)^{N-i}\left(s_{\boldsymbol{\sigma}^k(i)}(\boldsymbol{w}_{i-1}^k) - s_{\boldsymbol{\sigma}^k(i)}(\boldsymbol{w}_0^k)\right)\right\|^2}_{\triangleq B} \\ &\quad + \frac{2\mu^2}{1-t}\underbrace{\mathbb{E}\left\|\sum_{i=1}^N (I-\mu H)^{N-i}\xi(\boldsymbol{w}_{i-1}^k)\right\|^2}_{\triangleq C}\end{aligned} \tag{32}$$

for any $0 < t < 1$. Let us examine the terms in (32). To begin with:

$$A \overset{(18)}{\leq} \|I-\mu H\|^{2N}\|\widetilde{\boldsymbol{w}}_0^k\|^2 + \mu^2 \mathbb{E}\left\|\sum_{i=1}^N (I-\mu H)^{N-i} s_{\boldsymbol{\sigma}^k(i)}(\boldsymbol{w}_0^k)\right\|^2 \tag{33}$$

while

$$\begin{aligned}B &\overset{(a)}{\leq} N\sum_{i=1}^N \mathbb{E}\left\|(I-\mu H)^{N-i}\left(s_{\boldsymbol{\sigma}^k(i)}(\boldsymbol{w}_{i-1}^k) - s_{\boldsymbol{\sigma}^k(i)}(\boldsymbol{w}_0^k)\right)\right\|^2 \\ &\overset{(b)}{\leq} N\sum_{i=1}^N \|(I-\mu H)^{N-i}\|^2 \mathbb{E}\left\|s_{\boldsymbol{\sigma}^k(i)}(\boldsymbol{w}_{i-1}^k) - s_{\boldsymbol{\sigma}^k(i)}(\boldsymbol{w}_0^k)\right\|^2 \\ &\overset{(c)}{\leq} N\sum_{i=1}^N \mathbb{E}\left\|s_{\boldsymbol{\sigma}^k(i)}(\boldsymbol{w}_{i-1}^k) - s_{\boldsymbol{\sigma}^k(i)}(\boldsymbol{w}_0^k)\right\|^2\end{aligned} \tag{34}$$

where step (a) is due to Jensen's inequality:

$$\left\|\sum_{i=1}^N x_i\right\|^2 = N^2\left\|\sum_{i=1}^N \frac{1}{N}x_i\right\|^2 \leq N\sum_{i=1}^N \|x_i\|^2 \tag{35}$$

step (b) is due to the sub-multiplicative property of norms, and step (c) assumes a small enough μ so that

$$\|(I-\mu H)^{N-i}\|^2 \leq \left(\max\{1-\mu\nu, 1-\mu\delta\}\right)^{2N-2i} \leq 1 \tag{36}$$

With regards to the term involving the gradient noise difference in (34), we have:

$$\begin{aligned}&\|s_{\boldsymbol{\sigma}^k(i)}(\boldsymbol{w}_{i-1}^k) - s_{\boldsymbol{\sigma}^k(i)}(\boldsymbol{w}_0^k)\| \\ &\overset{(a)}{\leq} \|\nabla J(\boldsymbol{w}_{i-1}^k) - \nabla J(\boldsymbol{w}_0^k)\| \\ &\quad + \|\nabla Q(\boldsymbol{w}_{i-1}^k; x_{\boldsymbol{\sigma}^k(i)}) - \nabla Q(\boldsymbol{w}_0^k; x_{\boldsymbol{\sigma}^k(i)})\| \\ &\overset{(11)}{\leq} \delta\|\boldsymbol{w}_{i-1}^k - \boldsymbol{w}_0^k\| + \delta\|\boldsymbol{w}_{i-1}^k - \boldsymbol{w}_0^k\| \\ &\overset{(3)}{=} 2\delta\mu\left\|\sum_{n=1}^{i-1}\nabla Q(\boldsymbol{w}_{n-1}^k; x_{\boldsymbol{\sigma}^k(n)})\right\| \\ &\overset{(b)}{\leq} 2\delta\mu\sum_{n=1}^{i-1}\left(\delta\left\|\boldsymbol{w}_{n-1}^k - w^\star\right\| + \left\|\nabla Q(w^\star; x_{\boldsymbol{\sigma}^k(n)})\right\|\right)\end{aligned} \tag{37}$$

where (a) follows from the triangle inequality, and (b) follows from the triangle inequality and the Lipschitz assumption (11). To simplify the notation, we introduce the constant:

$$\mathcal{K} \triangleq \max_{n\in\{1,\ldots,N\}} \left\|\nabla Q(w^\star; x_{\boldsymbol{\sigma}^k(n)})\right\| \tag{38}$$

After substituting (37) into (34), we obtain:

$$\begin{aligned}B &\leq 4\delta^2\mu^2 N\sum_{i=1}^N \mathbb{E}\left(\sum_{n=1}^{i-1}\left\{\delta\left\|\boldsymbol{w}_{n-1}^k - w^\star\right\| + \mathcal{K}\right\}\right)^2 \\ &\overset{(35)}{\leq} 4\delta^2\mu^2 N\sum_{i=1}^N (i-1)\sum_{n=1}^{i-1}\left(2\delta^2\mathbb{E}\left\|\boldsymbol{w}_{n-1}^k - w^\star\right\|^2 + 2\mathcal{K}^2\right) \\ &\overset{(21)}{\leq} 4\delta^2\mu^2 N\sum_{i=1}^N (i-1)\sum_{n=1}^{i-1}\Big(O(\mu) + O(1)\Big), \quad k \gg 1\end{aligned}$$

$$= O(\mu^2), \quad k \gg 1 \tag{39}$$

Similarly, using the established stability of the algorithm in Lemma 1, we bound the third term after sufficient number of epochs:

$$\begin{aligned} C &\overset{(35)}{\leq} N\sum_{i=1}^{N} \|I-\mu H\|^{2N-2i}\mathbb{E}\,\|\xi(\boldsymbol{w}_{i-1}^k)\|^2 \\ &\overset{(15)}{\leq} N\kappa^2\sum_{i=1}^{N}\mathbb{E}\,\|\widetilde{\boldsymbol{w}}_{i-1}^k\|^4 \\ &\overset{(22)}{=} O(\mu^2), \quad k \gg 1 \end{aligned} \tag{40}$$

Substituting the three bounds (33), (39), and (40) into (32), we obtain for $k \gg 1$:

$$\begin{aligned} &\mathbb{E}\left[\|\widetilde{\boldsymbol{w}}_0^{k+1}\|^2 \,|\, \widetilde{\boldsymbol{w}}_0^k\right] \\ &\leq \frac{1}{t}\|I-\mu H\|^{2N}\|\widetilde{\boldsymbol{w}}_0^k\|^2 + \frac{\mu^2}{t}\mathbb{E}\left\|\sum_{i=1}^{N}(I-\mu H)^{N-i}s_{\boldsymbol{\sigma}^k(i)}(\boldsymbol{w}_0^k)\right\|^2 \\ &\quad + \frac{1}{1-t}O(\mu^4) \end{aligned} \tag{41}$$

Using the zero-sum property (30) of the random reshuffling scheme, it further follows that:

$$\begin{aligned} &\mathbb{E}\left\|\sum_{i=1}^{N}(I-\mu H)^{N-i}s_{\boldsymbol{\sigma}^k(i)}(\boldsymbol{w}_0^k)\right\|^2 \\ &\overset{(30)}{=} \mathbb{E}\left\|\sum_{i=1}^{N}(I-\mu H)^{N-i}s_{\boldsymbol{\sigma}^k(i)}(\boldsymbol{w}_0^k) - \underbrace{\sum_{i=1}^{N}s_{\boldsymbol{\sigma}^k(i)}(\boldsymbol{w}_0^k)}_{=0}\right\|^2 \\ &\overset{(a)}{=} \mathbb{E}\left\|\sum_{i=1}^{N}[(N-i)\mu H + O(\mu^2)]s_{\boldsymbol{\sigma}^k(i)}(\boldsymbol{w}_0^k)\right\|^2 \\ &\overset{(35)}{\leq} \mu^2 N\sum_{i=1}^{N}\mathbb{E}\left\|[(N-i)H+O(\mu)]s_{\boldsymbol{\sigma}^k(i)}(\boldsymbol{w}_0^k)\right\|^2 \\ &\leq \mu^2 N\sum_{i=1}^{N}\Big(2(N-i)^2\|H\|^2\mathbb{E}\left\|s_{\boldsymbol{\sigma}^k(i)}(\boldsymbol{w}_0^k)\right\|^2 \\ &\qquad + O(\mu^2)\mathbb{E}\,\|s_{\boldsymbol{\sigma}^k(i)}(\boldsymbol{w}_0^k)\|^2\Big) \end{aligned} \tag{42}$$

where step (a) uses the binomial expansion. Moreover, from (21) and (28) we can bound the variance of the gradient noise for $k \gg 1$ by:

$$\mathbb{E}\,\|s_{\boldsymbol{\sigma}^k(i)}(\boldsymbol{w}_{i-1}^k)\|^2 \leq \beta_e^2\mathbb{E}\,\|\widetilde{\boldsymbol{w}}_{i-1}^k\|^2 + \sigma_e^2 = O(1) \tag{43}$$

which allows us to conclude that

$$\mathbb{E}\left\|\sum_{i=1}^{N}(I-\mu H)^{N-i}s_{\boldsymbol{\sigma}^k(i)}(\boldsymbol{w}_0^k)\right\|^2 = O(\mu^2) \tag{44}$$

This fact is a critical improvement over traditional, independently sampled, gradient descent, where the zero-sum property (30) does not hold, causing (44) to be $O(1)$ instead. Setting $t = \|I-\mu H\|^N$, expression (41) implies that:

$$\begin{aligned} \mathbb{E}\,\|\widetilde{\boldsymbol{w}}_0^{k+1}\|^2 \leq \|I-\mu H\|^N\mathbb{E}\,\|\widetilde{\boldsymbol{w}}_0^k\|^2 + \frac{1}{\|I-\mu H\|^N}O(\mu^4) \\ + \frac{1}{1-\|I-\mu H\|^N}O(\mu^4) \end{aligned} \tag{45}$$

Note that, when μ is small enough, we have

$$\|I-\mu H\|^N \overset{(27)}{=} 1 - O(\mu) \tag{46}$$

so that

$$\limsup_{k\to\infty}\mathbb{E}\,\|\widetilde{\boldsymbol{w}}_0^k\|^2 \leq O(\mu^3) + O(\mu^2) = O(\mu^2) \tag{47}$$

□

3. EXPERIMENTS AND SIMULATIONS

In this section we illustrate the theoretical findings by means of numerical simulations. We consider the following logistic regression problem:

$$\min_{w} \quad J(w) = \frac{1}{N}\sum_{n=1}^{N}Q(w;h_n,\gamma(n)), \tag{48}$$

where $h_n \in \mathbb{R}^M$ is the feature vector, $\gamma(n) \in \{\pm 1\}$ is the label scalar, and

$$Q(w;h_n,\gamma_n) \triangleq \rho\|w\|^2 + \ln\left(1+\exp(-\gamma(n)h_n^{\mathsf{T}}w)\right). \tag{49}$$

The constant ρ is the regularization parameter. In the first simulation, we compare the performance of the standard stochastic gradient descent (SGD) algorithm (2)with replacement and the random reshuffling (RR) algorithm (3). In this simulation, we set $N = 1000$ and $M = 10$. Each h_n is generated from the normal distribution $\mathcal{N}(0;\Lambda_M)$, where Λ_M is a diagonal matrix with each diagonal entry generated from the uniform distribution $\mathcal{U}(1,10)$. To generate $\gamma(n)$, we first generate an auxiliary random vector $w_0 \in \mathbb{R}^M$ with each entry following $\mathcal{N}(0,1)$. Next, we generate $\boldsymbol{u}(n)$ from a uniform distribution $\mathcal{U}(0,1)$. If $\boldsymbol{u}(n) \leq 1/(1+\exp(-h_n^{\mathsf{T}}w_0))$ then $\gamma(n)$ is set as $+1$; otherwise $\gamma(n)$ is set as -1. We select $\rho = 0.1$ during all simulations. Figure 1 illustrates the MSD performance of the SGD and RR algorithms when $\mu = 0.003$. It is observed that the RR algorithm oscillates during the steady-state regime, and that the MSD at the $\boldsymbol{w}_0^k$ is the best among all iterates $\{\boldsymbol{w}_i^k\}_{i=1}^{N-1}$ during epoch k. Furthermore, it is also observed that RR has better MSD performance than SGD. Similar observations also occur in Fig. 2, where $\mu = 0.0003$. It is worth noting that the gap between SGD and RR is much larger in Fig. 2 than in Fig. 1. Since the steady-state MSD of standard SGD is on the order of $O(\mu)$, Fig. 2 implies that RR is on a higher order than $O(\mu)$.

Next, in the second simulation we verify the conclusion that the MSD for the starting point of each epoch for the random reshuffling algorithm, i.e., $\boldsymbol{w}_0^k$, can achieve $O(\mu^2)$ instead of $O(\mu)$. We still consider the regularized logistic regression problem (48) and (49), and the same experimental setting. In Lemma 2, we proved that

$$\limsup_{k\to\infty}\mathbb{E}\,\|\widetilde{\boldsymbol{w}}_0^k\|^2 \leq O(\mu^2), \tag{50}$$

which indicates that when μ is reduced by ten times, the MSD-performance $\mathbb{E}\,\|\widetilde{\boldsymbol{w}}_0^k\|^2$ should be improved by at least 20 dB. We observe a decay of about 20dB per decade in Fig. 3 for a logistic regression problem with $N = 25$ data points and 30dB per decade in Fig. 4 with $N = 1000$.

4. REFERENCES

[1] D. P. Bertsekas and J. N. Tsitsiklis, *Parallel and Distributed Computation: Numerical Methods*, Prentice Hall, NJ, 1989.

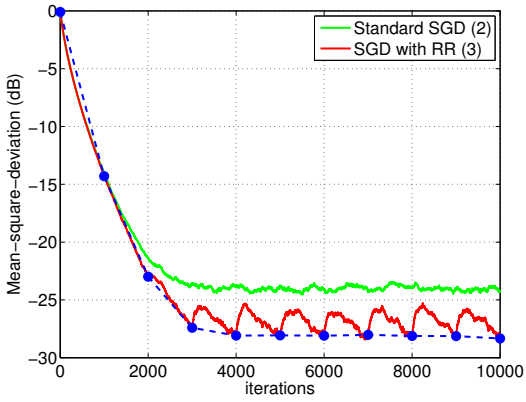

Fig. 1. RR has better MSD performance than standard SGD when $\mu = 0.003$. The dotted blue curve is drawn by connecting the MSD performance at the starting points of the successive epochs.

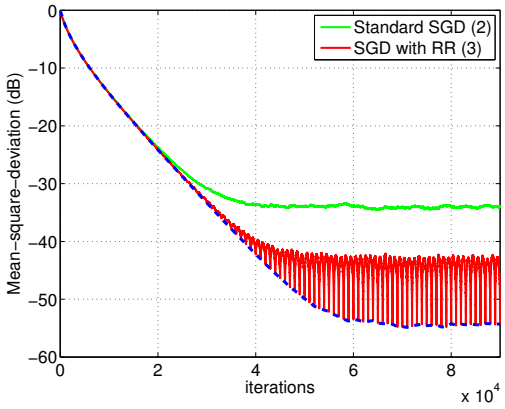

Fig. 2. RR has much better MSD performance than standard SGD when $\mu = 0.0003$. The dotted blue curve is drawn by connecting the MSD performance at the starting points of the successive epochs.

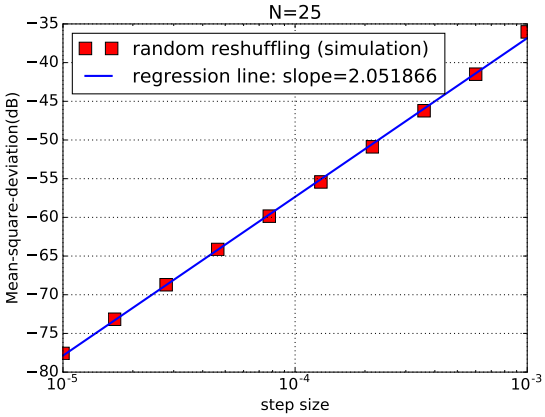

Fig. 3. Mean-square-deviation performance at steady-state versus the step size for a logistic problem involving $N = 25$ data points. The slope is around 20 dB per decade.

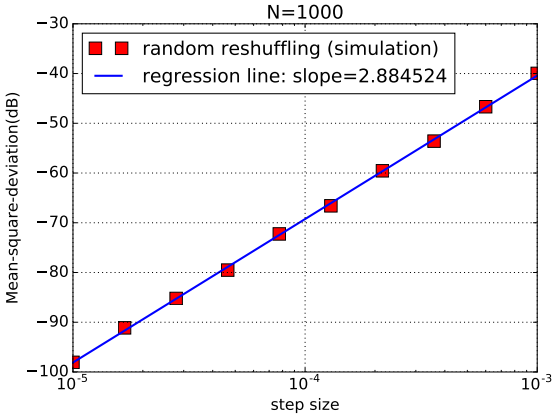

Fig. 4. Mean-square-deviation performance at steady-state versus the step size for a logistic problem involving $N = 1000$ data points. The slope is around 30 dB per decade.

[2] B. T. Polyak and A. B. Juditsky, "Acceleration of stochastic approximation by averaging," *SIAM Journal on Control and Optimization*, vol. 30, no. 4, pp. 838–855, 1992.

[3] B. T. Polyak, *Introduction to Optimization*, Optimization Software, New York, 1987.

[4] L. Bottou, "Large-scale machine learning with stochastic gradient descent," in *Proc. International Conference on Computational Statistics* (COMPSTAT), Paris, France, 2010, pp. 177–186.

[5] O. Bousquet and L. Bottou, "The tradeoffs of large scale learning," in *Proc. Advances in Neural Information Processing Systems* (NIPS), Vancouver, Canada, 2008, pp. 161–168.

[6] E. Moulines and F. R. Bach, "Non-asymptotic analysis of stochastic approximation algorithms for machine learning," in *Proc. Advances in Neural Information Processing Systems* (NIPS), Granada, Spain, 2011, pp. 451–459.

[7] T. Zhang, "Solving large scale linear prediction problems using stochastic gradient descent algorithms," in *Proc. International Conference on Machine Learning* (ICML), Canada, 2004, pp. 116–124.

[8] K. Yuan, B. Ying, S. Vlaski, and A. H. Sayed, "Stochastic gradient descent with finite samples sizes," in *Proc. IEEE International Workshop on Machine Learning for Signal Processing*, Salerno, Italy, 2016, pp. 1–6.

[9] L. Bottou, "Curiously fast convergence of some stochastic gradient descent algorithms," in *Proc. Symposium on Learning and Data Science*, Paris, 2009.

[10] B. Recht and C. Ré, "Toward a noncommutative arithmetic-geometric mean inequality: Conjectures, case-studies, and consequences," in *Proc. Conference On Learning Theory (COLT)*, 2012, pp. 1–11.

[11] M. Gürbüzbalaban, A. Ozdaglar, and P. Parrilo, "Why random reshuffling beats stochastic gradient descent," *arXiv:1510.08560*, Oct. 2015.

[12] D. P. Bertsekas, *Convex Optimization Algorithms*, Athena Scientific Belmont, 2015.

[13] O. Shamir, "Without-replacement sampling for stochastic gradient methods: Convergence results and application to distributed optimization," *arXiv:1603.00570*, Mar. 2016.

[14] A. H. Sayed, "Adaptation, learning, and optimization over networks," *Foundations and Trends in Machine Learning*, vol. 7, no. 4–5, pp. 311–801, 2014.