ISE Industrial and Systems Engineering

A Stochastic Trust Region Algorithm Based on Careful Step Normalization

Frank E. Curtis, Katya Scheinberg, and Rui Shi

Department of Industrial and Systems Engineering, Lehigh University

COR@L Technical Report 17T-17-R1

LEHIGH UNIVERSITY

COR@L COMPUTATIONAL OPTIMIZATION RESEARCH AT LEHIGH

A Stochastic Trust Region Algorithm Based on Careful Step Normalization

Frank E. Curtis*[1], Katya Scheinberg†[1], and Rui Shi‡[1]

[1]Department of Industrial and Systems Engineering, Lehigh University

Original Publication: January 27, 2018
Last Revised: June 26, 2018

Abstract

An algorithm is proposed for solving stochastic and finite sum minimization problems. Based on a trust region methodology, the algorithm employs normalized steps, at least as long as the norms of the stochastic gradient estimates are within a specified interval. The complete algorithm—which dynamically chooses whether or not to employ normalized steps—is proved to have convergence guarantees that are similar to those possessed by a traditional stochastic gradient approach under various sets of conditions related to the accuracy of the stochastic gradient estimates and choice of stepsize sequence. The results of numerical experiments are presented when the method is employed to minimize convex and nonconvex machine learning test problems. These results illustrate that the method can outperform a traditional stochastic gradient approach.

1 Introduction

The stochastic gradient (SG) method is the signature strategy for solving stochastic and finite-sum minimization problems. In this iterative approach, each step to update the solution estimate is obtained by taking a negative multiple of an unbiased gradient estimate. With careful choices for the stepsize sequence, the SG method possesses convergence guarantees and has been employed to great success for solving various types of problems, such as those arising in machine learning. For fundamental work on SG, see [19] and [20].

One disadvantage of the SG method is that stochastic gradients, like the gradients that they approximate, possess *no natural scaling*. By this, we mean that in order to guarantee convergence, the algorithm needs to choose stepsizes in a problem-dependent manner; e.g., common theoretical guarantees require that the stepsize is proportional to $1/L$, where L is a Lipschitz constant for the gradient of the objective function. This is in contrast to Newton's method for minimization, for which one can obtain (local) convergence guarantees with a stepsize of 1. Admittedly, Newton's method is not generally guaranteed to converge from remote starting points with unit stepsizes, but these observations do highlight a shortcoming of first-order methods, namely, that for convergence guarantees the stepsizes need always be chosen in a problem-dependent manner.

The purpose of this paper is to propose a new algorithm for stochastic and finite-sum minimization. Our proposed approach can be viewed as a modification of the SG method. The approach does not completely overcome the issue of requiring problem-dependent stepsizes, but we contend that our approach does, for practical purposes, reduce somewhat this dependence. This is achieved by employing, under certain conditions, *normalized* steps. We motivate our proposed approach by illustrating how it can be derived from a

*E-mail: frank.e.curtis@lehigh.edu
†E-mail: katyas@lehigh.edu
‡E-mail: rus415@lehigh.edu

trust region methodology. This work can be viewed as a first step toward designing new classes of first- and second-order trust region methods for solving stochastic and finite-sum minimization problems.

The use of normalized steps has previously been proposed in the context of (stochastic) gradient methods for solving minimization problems. For example, in a method that is similar to ours, [12] propose an approach that employs normalized steps in every iteration. They show that, if the objective function is *M-bounded* and *strictly-locally-quasi-convex*, the stochastic gradients are sufficiently accurate with respect to the true gradients (specifically, when mini-batch sizes are $\Omega(\epsilon^{-2})$), and a sufficiently large number of iterations are run (specifically, $\Omega(\epsilon^{-2})$), then their method will, with high probability, yield a solution estimate that is ϵ-optimal. By contrast, our approach, by employing a modified update that does not always involve the use of a normalized step, enjoys convergence guarantees under different assumptions. We argue in this paper that employing normalized steps in all iterations cannot lead to general convergence guarantees, which perhaps explains the additional assumptions required for convergence by [12].

It is also worthwhile to mention the broader literature. For important work on SG-type methods and their corresponding theoretical analyses, see, e.g., [1], [3], [6], [9], [10], [11], [13], and [18]. There are also numerous variants of SG methods based on gradient aggregation, iterative averaging, second-order techniques, momentum, acceleration, and beyond; for work on these, see [2] and the references therein. More related to our work are techniques that normalize steplengths based on *accumulated* gradient information; see, e.g., [7] and [21]. In a different direction, one should also contrast our work with stochastic trust region approaches, such as those in [16] and [5]. The approaches proposed in these papers, which are based on the use of randomized models of the objective function constructed during each iteration, are quite distinct from our proposed method. For example, these approaches follow a traditional trust region strategy of accepting or rejecting each step based on the magnitude of an (approximate) *actual-to-predicted reduction ratio*. Our method, on the other hand, is closer to the SG method in that it accepts the computed step in every iteration. Another distinction is that these other approaches rely on the use of so-called *fully linear* models of the objective function to obtain their convergence guarantees. Our convergence guarantees are obtained under straightforward upper bounds on the second moment of the stochastic gradient estimates, and do not require fully linear models.

The paper is organized as follows. Our algorithm and motivation for our specific iterate updating scheme are the subject of §2. In §3, we prove convergence guarantees for the algorithm under various types of assumptions on the stochastic gradient estimates and stepsize choices. The results of numerical experiments on test problems—some convex and some nonconvex—are given in §4. Concluding remarks are given in §5. All norms in the paper are Euclidean, i.e., $\|\cdot\| := \|\cdot\|_2$.

2 Algorithm

Our problem of interest is a stochastic optimization problem in which the goal is to minimize over a vector of decision variables, indicated by $x \in \mathbb{R}^n$, a function $f : \mathbb{R}^n \to \mathbb{R}$ defined by the expectation of another function $F : \mathbb{R}^n \times \Xi \to \mathbb{R}$ that depends on a random variable ξ, i.e.,

$$\min_{x \in \mathbb{R}^n} \ f(x) \ \text{ with } \ f(x) = \mathbb{E}_\xi[F(x,\xi)], \tag{2.1}$$

where $\mathbb{E}_\xi[\cdot]$ denotes expectation with respect to the distribution of ξ. Our algorithm is also applicable for finite-sum minimization where the objective takes the form

$$f(x) = \frac{1}{N}\sum_{i=1}^{N} f_i(x). \tag{2.2}$$

Such objectives often arise in sample average approximations of (2.1); e.g., see [22].

2.1 Algorithm Description

Our algorithm is stated below as TRish, a trust-region-*ish* algorithm for stochastic optimization. Each iteration involves taking a step along the negative of a stochastic gradient direction. In the context of problem (2.1), this stochastic gradient can be viewed as $g_k = \nabla_x F(x_k, \xi_k)$, where x_k is the current iterate and ξ_k is a realization of the random variable ξ. In the context of problem (2.2), it can be viewed as $g_k = \nabla_x f_{i_k}(x_k)$ where i_k has been chosen randomly as an index in $\{1, \dots, N\}$. In addition, in either case, g_k could represent an average of such quantities, i.e., over a set of independently generated realizations $\{\xi_{k,j}\}_{j \in \mathcal{S}_k}$ or over independently generated indices $\{i_{k,j}\}_{j \in \mathcal{S}_k}$. This leads to a so-called *mini-batch* approach with $\mathcal{S}_k$ representing the mini-batch of samples in the kth iteration. In the algorithm, we simply write $g_k \approx \nabla f(x_k)$ to cover all of these situations, since in any case g_k represents a stochastic gradient estimate for f at x_k.

Algorithm TRish (Trust-region-ish algorithm based on careful step normalization)

```
Choose an initial iterate $x_1$ and positive stepsizes $\{\alpha_k\}$.
Choose positive constants $\{\gamma_{1,k}\}$ and $\{\gamma_{2,k}\}$ such that $\gamma_{1,k} > \gamma_{2,k} > 0$ for all $k \in \mathbb{N}$.
for all $k \in \mathbb{N} := \{1, 2, \dots\}$ do
    Generate a stochastic gradient $g_k \approx \nabla f(x_k)$.
    Set
        $x_{k+1} \leftarrow x_k - \begin{cases} \gamma_{1,k}\alpha_k g_k & \text{if } \|g_k\| \in [0, \frac{1}{\gamma_{1,k}}) \\ \alpha_k g_k/\|g_k\| & \text{if } \|g_k\| \in [\frac{1}{\gamma_{1,k}}, \frac{1}{\gamma_{2,k}}] \\ \gamma_{2,k}\alpha_k g_k & \text{if } \|g_k\| \in (\frac{1}{\gamma_{2,k}}, \infty). \end{cases}$
end for
```

The scaling of the stochastic gradient employed in TRish can be motivated in the following manner. Given a stochastic gradient g_k and a stepsize α_k, consider the trust region subproblem

$$\min_{s \in \mathbb{R}^n} \; f(x_k) + g_k^T s \quad \text{s.t.} \quad \|s\| \leq \alpha_k. \tag{2.3}$$

The solution of this subproblem, namely, $s_k = -\alpha_k g_k/\|g_k\|$, represents the step that minimizes the first-order model $f(x_k) + g_k^T s$ of the objective function f at x_k subject to s having norm less than or equal to α_k. This is the prototypical strategy in a trust region methodology. When the norm of g_k falls within the interval $[\frac{1}{\gamma_{1,k}}, \frac{1}{\gamma_{2,k}}]$, TRish takes the step s_k. However, if this were to be done no matter the norm of g_k, then the resulting algorithm might fail to make progress in expectation. This is illustrated in the following example.

Example 2.1. *Suppose that, at a point $x_k \in \mathbb{R}$, one has $\nabla f(x_k) = 1$ and obtains*

$$g_k = \begin{cases} 6 \text{ with probability } \frac{1}{3} \\ -\frac{3}{2} \text{ with probability } \frac{2}{3}. \end{cases}$$

Then, $\mathbb{E}_k[g_k] = 1 = \nabla f(x_k)$, where $\mathbb{E}_k$ denotes expectation given that an algorithm has reached x_k as the kth iterate. However, this means that the normalized stochastic gradient satisfies

$$\frac{g_k}{\|g_k\|} = \begin{cases} 1 \text{ with probability } \frac{1}{3} \\ -1 \text{ with probability } \frac{2}{3}, \end{cases}$$

from which it follows that $s_k = -\alpha_k g_k/\|g_k\|$ is twice as likely to be a direction of ascent for f at x_k than it is to be a direction of descent for f at x_k.

One can argue from this example that, without potentially restrictive assumptions on the objective function f and/or the manner in which the stochastic gradient is computed, one cannot expect to be able to prove convergence guarantees for an algorithm that solely computes steps based on solving the trust region subproblem (2.3). In particular, the existence of any point (let alone more than one) at which the expectation is to follow an ascent direction foils the typical convergence theory for an SG approach; see, e.g., [2].

In TRish, we overcome the issue highlighted in Example 2.1 by only choosing the trust region step when the norm of the gradient is within a specified interval; otherwise, we compute a stochastic gradient step with a stepsize that is a multiple of α_k. It is for this reason that we refer to the algorithm as a trust-region-*ish* approach. Overall, as a function of the norm of the stochastic gradient, the norm of the step taken by the algorithm is illustrated in Figure 1. Note that care has been taken to make sure that the norm of the step is a continuous function of the norm of the stochastic gradient estimate. The plot in Figure 1 illustrates the relationship for moderate values of $(\gamma_{1,k}, \gamma_{2,k})$, but notice that for more extreme values (i.e., $\gamma_{1,k} \gg 0$ and $\gamma_{2,k} \approx 0$) the function would essentially be flat (except for stochastic gradients that are very small or large in norm), meaning that the stepsize would typically be scaled so that the step norm is approximately α_k for all $k \in \mathbb{N}$.

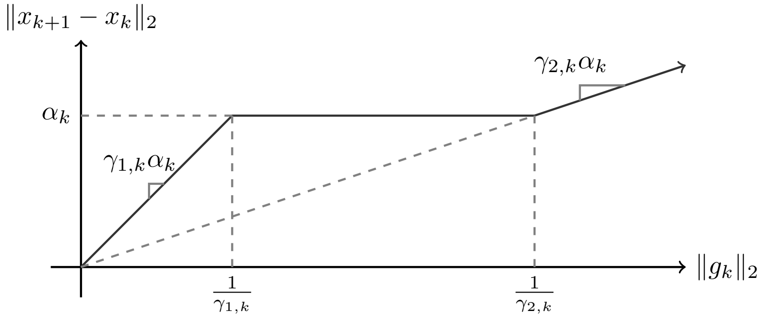

Figure 1: Relationship between $\|g_k\|$ and $\|x_{k+1} - x_k\|$ in Algorithm TRish.

Our convergence analysis in the next section requires certain restrictions on the choice of stepsizes—as is typical for (stochastic) gradient methods—and require certains restrictions on $\{\gamma_{1,k}\}$ and $\{\gamma_{2,k}\}$. For example, the issue in Example 2.1 is avoided as long as the pair $(\gamma_{1,k}, \gamma_{2,k})$ is chosen such that the step is not normalized with probability 1 at the given x_k, which means that—for this particular function, iterate, and variance in the stochastic gradient estimates—one cannot choose this pair such that $|6| \in [\frac{1}{\gamma_{1,k}}, \frac{1}{\gamma_{2,k}}]$ and $|\frac{3}{2}| \in [\frac{1}{\gamma_{1,k}}, \frac{1}{\gamma_{2,k}}]$ simultaneously. (In our convergence theory, this is avoided through upper bounds on the ratio $\frac{\gamma_{1,k}}{\gamma_{2,k}}$.) Various situations can illustrate how TRish avoids the issue in Example 2.1. For example, consider $\gamma_{1,k} = 1$ and $\gamma_{2,k} = \frac{1}{2}$, which leads to $\mathbb{E}_k[s_k] = -\frac{1}{2}\alpha_k(6)(\frac{1}{3}) - \alpha_k(-1)(\frac{2}{3}) = -\frac{1}{3}\alpha_k$, meaning that s_k is a descent direction in expectation. As another example, consider $\gamma_{1,k} = \frac{1}{4}$ and $\gamma_{2,k} = \frac{1}{6}$, which leads to $\mathbb{E}_k[s_k] = -\alpha_k(1)(\frac{1}{3}) - \frac{1}{6}\alpha_k(-\frac{3}{2})(\frac{2}{3}) = -\frac{1}{6}\alpha_k$, meaning again that s_k is a descent direction in expectation. Our theory reveals generic conditions that $\{(\gamma_{1,k}, \gamma_{2,k})\}$ must satisfy to attain different convergence properties for TRish. We also discuss, in §4, strategies for choosing these values in practice.

3 Convergence Analysis

Our goal in this section is to prove convergence guarantees for TRish that are similar to fundamental guarantees for a straightforward SG method; see, e.g., [2]. As in the notation for Example 2.1, our analysis uses $\mathbb{E}_k[\cdot]$ (resp. $\mathbb{P}_k[\cdot]$) to denote conditional expectation (resp. conditional probability) given that the algorithm has reached x_k as the kth iterate.

Throughout our analysis, we make the following assumption about the objective function.

Assumption 3.1. *The objective $f : \mathbb{R}^n \to \mathbb{R}$ is continuously differentiable and bounded below by $f_* = \inf_{x \in \mathbb{R}^n} f(x) \in \mathbb{R}$. In addition, at any $x \in \mathbb{R}^n$, the objective is bounded above by a first-order Taylor series* *approximation of f at x plus a quadratic term with constant $L \in (0,\infty)$, i.e.,*

$$f(x) \leq f(\overline{x}) + \nabla f(\overline{x})^T(x-\overline{x}) + \tfrac{1}{2}L\|x-\overline{x}\|^2 \quad \text{for all} \quad (x,\overline{x}) \in \mathbb{R}^n \times \mathbb{R}^n. \tag{3.1}$$

It is known that (3.1) holds if the gradient function ∇f is Lipschitz continuous with constant L. This is often referred to as L-smoothness of the function f.

We also make the following assumption about the stochastic gradients computed in TRish. This assumption is standard in analyses of SG methods; it is easily seen to be satisfied when the variance of the stochastic gradient estimate is uniformly bounded over $k \in \mathbb{N}$.

Assumption 3.2. *For all $k \in \mathbb{N}$, the stochastic gradient g_k is an unbiased estimator of $\nabla f(x_k)$ in the sense that $\mathbb{E}_k[g_k] = \nabla f(x_k)$. In addition, there exists a pair $(M_1, M_2) \in (0,\infty) \times (0,\infty)$ (independent of k) such that, for all $k \in \mathbb{N}$, the squared norm of g_k satisfies*

$$\mathbb{E}_k[\|g_k\|^2] \leq M_1 + M_2\|\nabla f(x_k)\|^2. \tag{3.2}$$

Under these assumptions, we prove the following lemma providing fundamental inequalities satisfied by TRish. For ease of reference in this result and throughout the remainder of our analysis, we define the following cases based on those indicated in Line 5 of TRish:

$$\text{“case 1”}: \quad \|g_k\| \in [0, \tfrac{1}{\gamma_{1,k}}); \quad \text{“case 2”}: \quad \|g_k\| \in [\tfrac{1}{\gamma_{1,k}}, \tfrac{1}{\gamma_{2,k}}]; \quad \text{“case 3”}: \quad \|g_k\| \in (\tfrac{1}{\gamma_{2,k}}, \infty).$$

The following lemma reveals an upper bound for the expected decrease in f for all $k \in \mathbb{N}$.

Lemma 3.1. *Under Assumptions 3.1 and 3.2, the iterates of TRish satisfy, for all $k \in \mathbb{N}$,*

$$\begin{aligned}\mathbb{E}_k[f(x_{k+1})] - f(x_k) \leq\ & -\gamma_{1,k}\alpha_k(1-\tfrac{1}{2}\gamma_{1,k}LM_2\alpha_k)\|\nabla f(x_k)\|^2 \\ & + (\gamma_{1,k}-\gamma_{2,k})\alpha_k\mathbb{P}_k[E_k]\mathbb{E}_k[\nabla f(x_k)^Tg_k|E_k] + \tfrac{1}{2}\gamma_{1,k}^2LM_1\alpha_k^2,\end{aligned} \tag{3.3}$$

where E_k is the event that $\nabla f(x_k)^Tg_k \geq 0$ and $\mathbb{P}_k[E_k]$ is the probability of this event.

Proof. Proof. For all $k \in \mathbb{N}$, let $s_k := x_{k+1} - x_k$ represent the step taken by the algorithm. By (3.1),

$$f(x_{k+1}) = f(x_k + s_k) \leq f(x_k) + \nabla f(x_k)^Ts_k + \tfrac{1}{2}L\|s_k\|^2.$$

Thus, with $C_{i,k}$ for $i \in \{1,2,3\}$ respectively representing the events that case 1, case 2, and case 3 occur, and with $\mathbb{P}_k[C_{i,k}]$ for $i \in \{1,2,3\}$ respectively representing the probabilities of these events, one finds from the law of total probability that

$$\begin{aligned}\mathbb{E}_k[f(x_{k+1})] - f(x_k) &\leq \mathbb{E}_k[\nabla f(x_k)^Ts_k] + \tfrac{1}{2}L\mathbb{E}_k[\|s_k\|^2] \\ &= \sum_{i=1}^3 \mathbb{P}_k[C_{i,k}]\mathbb{E}_k[\nabla f(x_k)^Ts_k|C_{i,k}] + \tfrac{1}{2}L\sum_{i=1}^3 \mathbb{P}_k[C_{i,k}]\mathbb{E}_k[\|s_k\|^2|C_{i,k}].\end{aligned} \tag{3.4}$$

In the event $C_{1,k}$, the algorithm yields $s_k = -\gamma_{1,k}\alpha_kg_k$, from which it follows that

$$\begin{aligned}&\mathbb{E}_k[\nabla f(x_k)^Ts_k|C_{1,k}] \\ =\ & -\gamma_{1,k}\alpha_k\mathbb{E}_k[\nabla f(x_k)^Tg_k|C_{1,k}] \\ =\ & -\gamma_{1,k}\alpha_k\mathbb{P}_k[E_k|C_{1,k}]\mathbb{E}_k[\nabla f(x_k)^Tg_k|C_{1,k}\cap E_k] - \gamma_{1,k}\alpha_k\mathbb{P}_k[\overline{E}_k|C_{1,k}]\mathbb{E}_k[\nabla f(x_k)^Tg_k|C_{1,k}\cap \overline{E}_k] \\ \leq\ & -\gamma_{2,k}\alpha_k\mathbb{P}_k[E_k|C_{1,k}]\mathbb{E}_k[\nabla f(x_k)^Tg_k|C_{1,k}\cap E_k] \\ & -\gamma_{1,k}\alpha_k(\mathbb{E}_k[\nabla f(x_k)^Tg_k|C_{1,k}] - \mathbb{P}_k[E_k|C_{1,k}]\mathbb{E}_k[\nabla f(x_k)^Tg_k|C_{1,k}\cap E_k]) \\ =\ & -\gamma_{1,k}\alpha_k\mathbb{E}_k[\nabla f(x_k)^Tg_k|C_{1,k}] + (\gamma_{1,k}-\gamma_{2,k})\alpha_k\mathbb{P}_k[E_k|C_{1,k}]\mathbb{E}_k[\nabla f(x_k)^Tg_k|C_{1,k}\cap E_k]\end{aligned} \tag{3.5}$$

along with the fact that

$$\mathbb{E}_k[\|s_k\|^2|C_{1,k}] = \gamma_{1,k}^2\alpha_k^2\mathbb{E}_k[\|g_k\|^2|C_{1,k}]. \tag{3.6}$$

In the event $C_{2,k}$, in which $\|g_k\|^{-1} \leq \gamma_{1,k}$ and $\|g_k\|^{-1} \geq \gamma_{2,k}$, one finds that

$$\begin{aligned}
&\mathbb{E}_k[\nabla f(x_k)^T s_k|C_{2,k}] \\
= &-\alpha_k\mathbb{E}_k\left[\frac{\nabla f(x_k)^T g_k}{\|g_k\|}\middle|C_{2,k}\right] \\
= &-\alpha_k\mathbb{P}_k[E_k|C_{2,k}]\mathbb{E}_k\left[\frac{\nabla f(x_k)^T g_k}{\|g_k\|}\middle|C_{2,k}\cap E_k\right] - \alpha_k\mathbb{P}_k[\overline{E}_k|C_{2,k}]\mathbb{E}_k\left[\frac{\nabla f(x_k)^T g_k}{\|g_k\|}\middle|C_{2,k}\cap \overline{E}_k\right] \\
\leq &-\gamma_{2,k}\alpha_k\mathbb{P}_k[E_k|C_{2,k}]\mathbb{E}_k[\nabla f(x_k)^T g_k|C_{2,k}\cap E_k] - \gamma_{1,k}\alpha_k\mathbb{P}_k[\overline{E}_k|C_{2,k}]\mathbb{E}_k[\nabla f(x_k)^T g_k|C_{2,k}\cap \overline{E}_k] \\
= &-\gamma_{2,k}\alpha_k\mathbb{P}_k[E_k|C_{2,k}]\mathbb{E}_k[\nabla f(x_k)^T g_k|C_{2,k}\cap E_k] \\
&-\gamma_{1,k}\alpha_k(\mathbb{E}_k[\nabla f(x_k)^T g_k|C_{2,k}] - \mathbb{P}_k[E_k|C_{2,k}]\mathbb{E}_k[\nabla f(x_k)^T g_k|C_{2,k}\cap E_k]) \\
= &-\gamma_{1,k}\alpha_k\mathbb{E}_k[\nabla f(x_k)^T g_k|C_{2,k}] + (\gamma_{1,k}-\gamma_{2,k})\alpha_k\mathbb{P}_k[E_k|C_{2,k}]\mathbb{E}_k[\nabla f(x_k)^T g_k|C_{2,k}\cap E_k]
\end{aligned} \tag{3.7}$$

along with the fact that

$$\mathbb{E}_k[\|s_k\|^2|C_{2,k}] = \alpha_k^2 \leq \gamma_{1,k}^2\alpha_k^2\mathbb{E}_k[\|g_k\|^2|C_{2,k}]. \tag{3.8}$$

In the event $C_{3,k}$, the algorithm yields $s_k = -\gamma_{2,k}\alpha_k g_k$, from which it follows that

$$\begin{aligned}
&\mathbb{E}_k[\nabla f(x_k)^T s_k|C_{3,k}] \\
= &-\gamma_{2,k}\alpha_k\mathbb{E}_k[\nabla f(x_k)^T g_k|C_{3,k}] \\
\leq &-\gamma_{2,k}\alpha_k\mathbb{P}_k[E_k|C_{3,k}]\mathbb{E}_k[\nabla f(x_k)^T g_k|C_{3,k}\cap E_k] - \gamma_{1,k}\alpha_k\mathbb{P}_k[\overline{E}_k|C_{3,k}]\mathbb{E}_k[\nabla f(x_k)^T g_k|C_{3,k}\cap \overline{E}_k] \\
= &-\gamma_{2,k}\alpha_k\mathbb{P}_k[E_k|C_{3,k}]\mathbb{E}_k[\nabla f(x_k)^T g_k|C_{3,k}\cap E_k] \\
&-\gamma_{1,k}\alpha_k(\mathbb{E}_k[\nabla f(x_k)^T g_k|C_{3,k}] - \mathbb{P}_k[E_k|C_{3,k}]\mathbb{E}_k[\nabla f(x_k)^T g_k|C_{3,k}\cap E_k]) \\
= &-\gamma_{1,k}\alpha_k\mathbb{E}_k[\nabla f(x_k)^T g_k|C_{3,k}] + (\gamma_{1,k}-\gamma_{2,k})\alpha_k\mathbb{P}_k[E_k|C_{3,k}]\mathbb{E}_k[\nabla f(x_k)^T g_k|C_{3,k}\cap E_k]
\end{aligned} \tag{3.9}$$

along with the fact that

$$\mathbb{E}_k[\|s_k\|^2|C_{3,k}] = \gamma_{2,k}^2\alpha_k^2\mathbb{E}_k[\|g_k\|^2|C_{3,k}] \leq \gamma_{1,k}^2\alpha_k^2\mathbb{E}_k[\|g_k\|^2|C_{3,k}]. \tag{3.10}$$

Combining (3.4)–(3.10), it follows that

$$\begin{aligned}
&\mathbb{E}_k[f(x_{k+1})] - f(x_k) \\
\leq &-\gamma_{1,k}\alpha_k\|\nabla f(x_k)\|^2 + (\gamma_{1,k}-\gamma_{2,k})\alpha_k\mathbb{P}_k[E_k]\mathbb{E}_k[\nabla f(x_k)^T g_k|E_k] + \tfrac{1}{2}\gamma_{1,k}^2 L\alpha_k^2\mathbb{E}_k[\|g_k\|^2].
\end{aligned}$$

Applying the upper bound for the last term in (3.2) and rearranging terms yields the result. □

For some (but not all) of our convergence guarantees, we also make the following assumption.

Assumption 3.3. *At any $x \in \mathbb{R}^n$, the Polyak-Lojasiewicz condition holds with $c \in (0,\infty)$, i.e.,*

$$2c(f(x) - f_*) \leq \|\nabla f(x)\|^2 \ \ \textit{for all} \ \ x \in \mathbb{R}^n. \tag{3.11}$$

Assumptions 3.1 and 3.3 do not ensure that a stationary point for f exists, though, when combined, they do guarantee that any stationary point for f is a global minimizer of f. Assumption 3.3 holds when f is c-strongly convex, but it is also satisfied for other functions that are not convex. We direct the interested reader to [14] for a discussion on the relationship between the Polyak-Łojasiewicz condition and the related *error bounds*, *essential strong convexity*, *weak strong convexity*, *restricted secant inequality*, and *quadratic growth* conditions. In short, when f has a Lipschitz continuous gradient, the Polyak-Łojasiewicz is the weakest of these except for the quadratic growth condition, though these two are equivalent when f is convex.

We now proceed to prove convergence guarantees for TRish in various cases depending on whether or not the Polyak-Łojasiewicz condition (hereafter referred to as the P-L condition) holds and based on different sets of properties of the sequence of stepsizes and stochastic gradient estimates. Our analysis covers various types of convex and nonconvex objective functions.

3.1 P-L Condition and Constant Parameters

Let us first prove a convergence result for TRish when the P-L condition holds and each sequence $\{\alpha_k\}$, $\{\gamma_{1,k}\}$, and $\{\gamma_{2,k}\}$ is constant. This result appears in this section as Theorem 3.1.

Our first requirement toward proving Theorem 3.1 is the following lemma.

Lemma 3.2. *Under Assumption 3.2, it follows that, for all $k \in \mathbb{N}$,*

$$\mathbb{P}_k[E_k]\mathbb{E}_k[\nabla f(x_k)^T g_k|E_k] \leq h_1 + h_2\|\nabla f(x_k)\|^2 \tag{3.12}$$

for any $(h_1, h_2) \in (0,\infty) \times (0,\infty)$ such that $h_1 \geq \frac{1}{2}\sqrt{M_1}$ and $h_2 \geq \frac{1}{2}\sqrt{M_1} + \sqrt{M_2}$.

Proof. Proof. One finds with the law of total probability that

$$\begin{aligned}\mathbb{P}_k[E_k]\mathbb{E}_k[\nabla f(x_k)^T g_k|E_k] &\leq \mathbb{P}_k[E_k]\mathbb{E}_k[\|\nabla f(x_k)\|\|g_k\||E_k] \\ &= \|\nabla f(x_k)\|(\mathbb{P}_k[E_k]\mathbb{E}_k[\|g_k\||E_k]) \\ &= \|\nabla f(x_k)\|(\mathbb{E}_k[\|g_k\|] - \mathbb{P}_k[\overline{E}_k]\mathbb{E}_k[\|g_k\||\overline{E}_k]) \\ &\leq \|\nabla f(x_k)\|\mathbb{E}_k[\|g_k\|].\end{aligned}$$

Then, by Jensen's Inequality, concavity of the square root, and Assumption 3.2, one finds that

$$\mathbb{E}_k[\|g_k\|] \leq \sqrt{\mathbb{E}_k[\|g_k\|^2]} \leq \sqrt{M_1 + M_2\|\nabla f(x_k)\|^2} \leq \sqrt{M_1} + \sqrt{M_2}\|\nabla f(x_k)\|.$$

Therefore, by combining the inequalities above, one finds that

$$\begin{aligned}\mathbb{P}_k[E_k]\mathbb{E}_k[\nabla f(x_k)^T g_k|E_k] &\leq \|\nabla f(x_k)\|(\sqrt{M_1} + \sqrt{M_2}\|\nabla f(x_k)\|) \\ &= \sqrt{M_1}\|\nabla f(x_k)\| + \sqrt{M_2}\|\nabla f(x_k)\|^2 \\ &\leq \frac{1}{2}\sqrt{M_1}(1 + \|\nabla f(x_k)\|^2) + \sqrt{M_2}\|\nabla f(x_k)\|^2 \\ &= \frac{1}{2}\sqrt{M_1} + \left(\frac{1}{2}\sqrt{M_1} + \sqrt{M_2}\right)\|\nabla f(x_k)\|^2,\end{aligned}$$

where the second inequality follows by the fact that $a \leq \frac{1}{2}(1 + a^2)$ for any $a \in \mathbb{R}$. □

While the upper bound on $\mathbb{E}_k[\|g_k\|^2]$ stated as (3.2) in Assumption 3.2 is standard in the literature, the quantity on the left-hand side of (3.12)—which Lemma 3.2 shows is bounded in a similar manner—is uniquely important for our analysis. For this reason, we feel that it is useful to provide specific examples illustrating how this quantity is bounded. We state two related examples next.

Example 3.1. *Suppose $f : \mathbb{R} \to \mathbb{R}$ and x_k are given such that $\nabla f(x_k) = \mu_k \in \mathbb{R}$, where without loss of generality one can assume that $\mu_k \geq 0$. In addition, suppose that g_k follows a normal distribution with mean μ_k and variance σ_k^2. Then,*

$$\begin{aligned}\mathbb{P}_k[E_k]\mathbb{E}_k[\nabla f(x_k)^T g_k|E_k] &= \mu_k \int_0^\infty g \tfrac{1}{\sqrt{2\pi}\sigma_k} e^{\frac{-(g-\mu_k)^2}{2\sigma_k^2}}\, dg \\ &= \mu_k \int_0^{\mu_k} g \tfrac{1}{\sqrt{2\pi}\sigma_k} e^{\frac{-(g-\mu_k)^2}{2\sigma_k^2}}\, dg + \mu_k \int_{\mu_k}^\infty g \tfrac{1}{\sqrt{2\pi}\sigma_k} e^{\frac{-(g-\mu_k)^2}{2\sigma_k^2}}\, dg.\end{aligned}$$

Let us separately investigate these two terms on the right-hand side. First, one finds that

$$\mu_k \int_0^{\mu_k} g \tfrac{1}{\sqrt{2\pi}\sigma_k} e^{\frac{-(g-\mu_k)^2}{2\sigma_k^2}}\, dg \leq \mu_k^2 \int_0^{\mu_k} \tfrac{1}{\sqrt{2\pi}\sigma_k} e^{\frac{-(g-\mu_k)^2}{2\sigma_k^2}}\, dg \leq \mu_k^2 \int_{-\infty}^{\mu_k} \tfrac{1}{\sqrt{2\pi}\sigma_k} e^{\frac{-(g-\mu_k)^2}{2\sigma_k^2}}\, dg = \tfrac{1}{2}\mu_k^2.$$

Second, one finds that

$$\begin{aligned}\mu_k \int_{\mu_k}^{\infty} g \tfrac{1}{\sqrt{2\pi}\sigma_k} e^{\frac{-(g-\mu_k)^2}{2\sigma_k^2}} dg &= \mu_k \int_0^{\infty} (t+\mu_k) \tfrac{1}{\sqrt{2\pi}\sigma_k} e^{\frac{-t^2}{2\sigma_k^2}} dt \\ &= \mu_k \int_0^{\infty} t \tfrac{1}{\sqrt{2\pi}\sigma_k} e^{\frac{-t^2}{2\sigma_k^2}} dt + \mu_k^2 \int_0^{\infty} \tfrac{1}{\sqrt{2\pi}\sigma_k} e^{\frac{-t^2}{2\sigma_k^2}} dt = \mu_k \tfrac{\sigma_k}{\sqrt{2\pi}} + \tfrac{1}{2}\mu_k^2.\end{aligned}$$

Thus, combining the bounds above, one finds that

$$\mathbb{P}_k[E_k]\mathbb{E}_k[\nabla f(x_k)^T g_k | E_k] \leq \mu_k \frac{\sigma_k}{\sqrt{2\pi}} + \mu_k^2 \leq \left(\frac{\mu_k^2+1}{2}\right)\frac{\sigma_k}{\sqrt{2\pi}} + \mu_k^2 = \frac{\sigma_k}{2\sqrt{2\pi}} + \left(1 + \frac{\sigma_k}{2\sqrt{2\pi}}\right)\mu_k^2.$$

Overall, if $\sigma_k \leq \sigma$ for some positive $\sigma \in \mathbb{R}$ for all $k \in \mathbb{N}$, then (3.12) holds with

$$h_1 = \frac{\sigma}{2\sqrt{2\pi}} \quad \text{and} \quad h_2 = 1 + \frac{\sigma}{2\sqrt{2\pi}}. \tag{3.13}$$

Example 3.2. *Suppose $f : \mathbb{R}^n \to \mathbb{R}$ and x_k are given such that $\nabla f(x_k) = \mu_k \in \mathbb{R}^n$. In addition, suppose that g_k follows a normal distribution with mean μ_k and covariance matrix Σ_k. Then, by Theorem 3.3.3 in [23], the inner product $\nabla f(x_k)^T g_k$ follows a normal distribution with mean $\|\mu_k\|^2$ and variance $\mu_k^T \Sigma_k \mu_k$. Hence, following the analysis in Example 3.1, if $\sqrt{\mu_k^T \Sigma_k \mu_k} \leq \sigma$ for some positive $\sigma \in \mathbb{R}$ for all $k \in \mathbb{N}$, then (3.12) holds with h_1 and h_2 from (3.13).*

We now prove our first theorem on the behavior of TRish.

Theorem 3.1. *Under Assumptions 3.1, 3.2, and 3.3, and with a pair (h_1, h_2) satisfying the inequalities in Lemma 3.2, suppose that TRish is run with $(\gamma_{1,k}, \gamma_{2,k}) = (\gamma_1, \gamma_2)$ for all $k \in \mathbb{N}$ such that $\frac{\gamma_1}{\gamma_2} < \frac{h_2}{h_2 - 1}$ (meaning $\gamma_1 - h_2(\gamma_1 - \gamma_2) > 0$) and with $\alpha_k = \alpha$ for all $k \in \mathbb{N}$ such that*

$$0 < \alpha \leq \min\left\{\frac{1}{2c\theta_1}, \frac{\gamma_1 - h_2(\gamma_1 - \gamma_2)}{\gamma_1 L M_2}\right\}, \tag{3.14}$$

where

$$\theta_1 = \tfrac{1}{2}(\gamma_1 - h_2(\gamma_1 - \gamma_2)) > 0. \tag{3.15}$$

Then, for all $k \in \mathbb{N}$, the expected optimality gap satisfies

$$\mathbb{E}[f(x_{k+1})] - f_* \leq \frac{\theta_2}{2c\alpha\theta_1} + (1 - 2c\alpha\theta_1)^{k-1}\left(f(x_1) - f_* - \frac{\theta_2}{2c\alpha\theta_1}\right) \xrightarrow{k\to\infty} \frac{\theta_2}{2c\alpha\theta_1}, \tag{3.16}$$

where

$$\theta_2 = h_1(\gamma_1 - \gamma_2)\alpha + \tfrac{1}{2}\gamma_1^2 L M_1 \alpha^2 > 0. \tag{3.17}$$

Proof. Proof. Combining the results of Lemmas 3.1 and 3.2, it follows that, for all $k \in \mathbb{N}$,

$$\begin{aligned}\mathbb{E}_k[f(x_{k+1})] - f(x_k) \leq & -\gamma_1\alpha(1 - \tfrac{1}{2}\gamma_1 L M_2 \alpha)\|\nabla f(x_k)\|^2 \\ & + (\gamma_1 - \gamma_2)\alpha(h_1 + h_2\|\nabla f(x_k)\|^2) + \tfrac{1}{2}\gamma_1^2 L M_1 \alpha^2.\end{aligned} \tag{3.18}$$

Therefore, with (θ_1, θ_2) defined in (3.15)/(3.17), it follows with (3.11) that, for all $k \in \mathbb{N}$,

$$\begin{aligned}\mathbb{E}_k[f(x_{k+1})] - f(x_k) &\leq -\alpha\theta_1\|\nabla f(x_k)\|^2 + \theta_2 \\ &\leq -2c\alpha\theta_1(f(x_k) - f_*) + \theta_2.\end{aligned}$$

Adding and subtracting f_* on the left-hand side, taking total expectations, and rearranging yields

$$\begin{aligned}\mathbb{E}[f(x_{k+1})] - f_* &\leq (1 - 2c\alpha\theta_1)(\mathbb{E}[f(x_k)] - f_*) + \theta_2 \\ &= \frac{\theta_2}{2c\alpha\theta_1} + (1 - 2c\alpha\theta_1)(\mathbb{E}[f(x_k)] - f_*) + \theta_2 - \frac{\theta_2}{2c\alpha\theta_1} \\ &= \frac{\theta_2}{2c\alpha\theta_1} + (1 - 2c\alpha\theta_1)\left(\mathbb{E}[f(x_k)] - f_* - \frac{\theta_2}{2c\alpha\theta_1}\right).\end{aligned}$$

Since $1 - 2c\alpha\theta_1 \in (0, 1)$, this represents a contraction inequality. Applying the result repeatedly through iteration $k \in \mathbb{N}$, one obtains the desired result. □

It is worthwhile to compare the result of Theorem 3.1 with a corresponding result known to hold for a straightforward SG method. For example, from [2, Thm. 4.6] with our notation, it is known that for an SG method with fixed stepsize $\alpha = \frac{1}{LM_2}$ an upper bound for the expected optimality gap converges to $\frac{\alpha L M_1}{2c} = \frac{M_1}{2cM_2}$. On the other hand, the analysis in Theorem 3.1 shows that TRish with $\alpha = \frac{\gamma_1 - h_2(\gamma_1 - \gamma_2)}{\gamma_1 L M_2}$ (which may occur, e.g., if $c \approx 0$) yields an upper bound for the expected optimality gap that converges to

$$\frac{h_1(\gamma_1 - \gamma_2) + \frac{1}{2}\gamma_1^2 L M_1 \alpha}{c(\gamma_1 - h_2(\gamma_1 - \gamma_2))} = \frac{h_1(\gamma_1 - \gamma_2)}{c(\gamma_1 - h_2(\gamma_1 - \gamma_2))} + \frac{\gamma_1 M_1}{2cM_2}. \tag{3.19}$$

We can now make a couple of observations. On one hand, if $h_1 \approx \frac{1}{2}\sqrt{M_1}$ and $h_2 \approx M_2 \approx 1$, then the condition that $\frac{\gamma_1}{\gamma_2} < \frac{h_2}{h_2 - 1}$ essentially does not restrict (γ_1, γ_2), in which case (3.19) is approximately

$$\frac{\sqrt{M_1}(\gamma_1 - \gamma_2)}{2c\gamma_2} + \frac{\gamma_1 M_1}{2c}.$$

This quantity is less than $\frac{M_1}{2c}$, i.e., the approximate bound for SG, if, e.g., the parameters satisfy $\gamma_1 \in (0, 1)$ with $\gamma_2 \geq \frac{\gamma_1}{1 + (1 - \gamma_1)\sqrt{M_1}} \in (0, \gamma_1)$. On the other hand, if $h_1 \approx \frac{1}{2}\sqrt{M_1}$ and $h_2 \approx \frac{1}{2}\sqrt{M_1} + \sqrt{M_2}$ with $M_1 \gg 0$, then the condition that $\frac{\gamma_1}{\gamma_2} < \frac{h_2}{h_2 - 1}$ essentially requires that $\gamma_1 \approx \gamma_2$, in which case the bound (3.19) is approximately $\frac{\gamma_1 M_1}{2cM_2}$, which is less than the bound for SG if $\gamma_1 \in (0, 1)$. Overall, while we are not necessarily recommending that one employes TRish with the parameter settings mentioned in this discussion, we have at least been able to demonstrate in both of these cases that TRish can possess an asymptotic bound on the expected optimality gap that is on par with that for SG. (For a detailed discussion on how to choose (γ_1, γ_2) in practice, see §4.1.)

Besides the conclusions of the previous paragraph, the result of Theorem 3.1 points to fundamental differences between TRish and SG for certain choices of the input parameters. In particular, the result in [2, Thm. 4.6] points to a well-known trade-off for SG with a fixed stepsize: If a relatively large stepsize is employed, then the rate to achieve the asymptotic expected optimality gap involves a better constant at the sake of the upper bound on the gap being relatively large, i.e., $\frac{\alpha L M_1}{2c}$, which is proportional to the stepsize α. On the other hand, one can achieve a smaller upper bound on the expected optimality gap with a smaller α, but at the cost of a worse constant in the rate to achieve that gap. A similar conclusion can be derived from (3.16) for TRish: One can control the constant $(1 - 2c\alpha\theta_1)$ by the choice of α. However, the effect of α on the expected optimality gap is not exactly the same for TRish as for an SG method. This can be seen in the fact that the left-hand side of (3.19) involves one term that decreases with α, but another term that does not. That said, one can compensate for this in TRish if one ties the difference $\gamma_1 - \gamma_2$ to the stepsize α. This idea can be seen in the first of our two theorems in the next subsection.

3.2 P-L Condition and Sublinearly Diminishing Stepsizes

Let us now consider the behavior of TRish when the P-L condition holds and diminishing stepsizes are employed. Our first theorem in this setting, which makes the same assumptions as Theorem 3.1, but involves different parameter choices, is the following. (The parameter choices in the theorem could be generalized even further. However, we have made certain choices—e.g., to have $\{\gamma_1\}$ be constant—for some amount of simplicity in the proof while still maintaining generality. One could prove a similar result with $\{\gamma_2\}$ constant instead, or with neither $\{\gamma_1\}$ nor $\{\gamma_2\}$ constant, as long as the sequence $\{\gamma_{1,k} - \gamma_{2,k}\}$ is proportional to α_k, as it is in the following theorem.)

Theorem 3.2. *Under Assumptions 3.1, 3.2, and 3.3, and with a pair (h_1, h_2) satisfying the inequalities in Lemma 3.2, suppose that TRish is run with $\gamma_{1,k} = \gamma_1 > 0$, $\gamma_{2,k} = \gamma_1(1 - \frac{1}{2}\eta\alpha_k)$ for $\eta \in (0,1)$, and*

$$\alpha_k = \frac{a}{b+k} \text{ for some } a \in \left(\frac{1}{c\gamma_1}, \frac{b+1}{c\gamma_1}\right) \text{ and } b > 0 \text{ with } \alpha_1 \in \left(0, \min\left\{\frac{1}{\eta}, \frac{1}{\eta h_2 + \gamma_1 L M_2}\right\}\right] \tag{3.20}$$

for all $k \in \mathbb{N}$. Then, for all $k \in \mathbb{N}$, the expected optimality gap satisfies

$$\mathbb{E}[f(x_k)] - f_* \leq \frac{\phi}{b+k}, \tag{3.21}$$

where

$$\phi = \max\left\{\frac{a^2\delta}{ac\gamma_1 - 1}, (b+1)(f(x_1) - f_*)\right\} > 0 \tag{3.22}$$

$$\text{and} \quad \delta = \tfrac{1}{2}\gamma_1(\eta h_1 + \gamma_1 L M_1) > 0. \tag{3.23}$$

Proof. Proof. First observe that the restrictions on $\{\alpha_k\}$ in (3.20) ensure that $\gamma_{2,k} > 0$, $\gamma_1 - \gamma_{2,k} = \frac{1}{2}\gamma_1\eta\alpha_k$, and $1 - \frac{1}{2}(\eta h_2 + \gamma_1 L M_2)\alpha_k \geq \frac{1}{2}$ for all $k \in \mathbb{N}$. Thus, similar to the proof of Theorem 3.1, for all $k \in \mathbb{N}$,

$$\begin{aligned}
\mathbb{E}_k[f(x_{k+1})] - f(x_k) \leq &-\gamma_1\alpha_k(1 - \tfrac{1}{2}\gamma_1 L M_2\alpha_k)\|\nabla f(x_k)\|^2 \\
&+ (\gamma_1 - \gamma_{2,k})\alpha_k(h_1 + h_2\|\nabla f(x_k)\|^2) + \tfrac{1}{2}\gamma_1^2 L M_1\alpha_k^2 \\
= &-\gamma_1\alpha_k(1 - \tfrac{1}{2}\gamma_1 L M_2\alpha_k)\|\nabla f(x_k)\|^2 \\
&+ \tfrac{1}{2}\gamma_1\eta\alpha_k^2(h_1 + h_2\|\nabla f(x_k)\|^2) + \tfrac{1}{2}\gamma_1^2 L M_1\alpha_k^2 \\
= &-\gamma_1\alpha_k(1 - \tfrac{1}{2}(\eta h_2 + \gamma_1 L M_2)\alpha_k)\|\nabla f(x_k)\|^2 + \tfrac{1}{2}\gamma_1(\eta h_1 + \gamma_1 L M_1)\alpha_k^2 \\
\leq &-\tfrac{1}{2}\gamma_1\alpha_k\|\nabla f(x_k)\|^2 + \tfrac{1}{2}\gamma_1(\eta h_1 + \gamma_1 L M_1)\alpha_k^2.
\end{aligned}$$

Therefore, with δ defined in (3.23), it follows with (3.11) that, for all $k \in \mathbb{N}$,

$$\begin{aligned}
\mathbb{E}_k[f(x_{k+1})] - f(x_k) &\leq -\tfrac{1}{2}\gamma_1\alpha_k\|\nabla f(x_k)\|^2 + \delta\alpha_k^2 \\
&\leq -c\gamma_1\alpha_k(f(x_k) - f_*) + \delta\alpha_k^2.
\end{aligned} \tag{3.24}$$

Adding and subtracting f_* on the left-hand side, taking total expectations, and rearranging yields

$$\mathbb{E}[f(x_{k+1})] - f_* \leq (1 - c\gamma_1\alpha_k)(\mathbb{E}[f(x_k)] - f_*) + \delta\alpha_k^2. \tag{3.25}$$

Let us now prove (3.21) by induction. First, for $k = 1$, the inequality holds by the definition of ϕ. Now suppose that (3.21) holds up to $k \in \mathbb{N}$; then, for $k+1$, one finds from above that

$$\begin{aligned}\mathbb{E}[f(x_{k+1})] - f_* &\leq (1 - c\gamma_1\alpha_k)(\mathbb{E}[f(x_k)] - f_*) + \delta\alpha_k^2 \\ &= \left(1 - \frac{ac\gamma_1}{b+k}\right)(\mathbb{E}[f(x_k)] - f_*) + \frac{a^2\delta}{(b+k)^2} \\ &\leq \left(1 - \frac{ac\gamma_1}{b+k}\right)\frac{\phi}{b+k} + \frac{a^2\delta}{(b+k)^2} \\ &= \frac{(b+k)\phi}{(b+k)^2} - \frac{ac\gamma_1\phi}{(b+k)^2} + \frac{a^2\delta}{(b+k)^2} \\ &= \frac{(b+k-1)\phi}{(b+k)^2} - \frac{(ac\gamma_1 - 1)\phi}{(b+k)^2} + \frac{a^2\delta}{(b+k)^2} \\ &\leq \frac{(b+k-1)\phi}{(b+k)^2} \leq \frac{\phi}{b+k+1},\end{aligned}$$

where the last two inequalities follow from the definition of ϕ and since $(b+k-1)(b+k+1) \leq (b+k)^2$, respectively. The desired conclusion now follows from this inductive argument. □

As one might predict from the discussion at the end of §3.1, in Theorem 3.2 we have been able to prove sublinear convergence of the expected optimality gap by tying the rate that $\{\gamma_{1,k} - \gamma_{2,k}\}$ vanishes to the rate that $\{\alpha_k\}$ vanishes; in particular, both the differences and the stepsizes diminish sublinearly, as is the case in similar results for SG methods.

One might also be interested in the behavior of TRish when the sequences $\{\gamma_{1,k}\}$ and $\{\gamma_{2,k}\}$ are constant while only the stepsizes decrease sublinearly. For example, this might be of interest since otherwise there are additional parameters to estimate and/or to tune. In the remainder of this subsection, we prove a sublinear convergence result under this setting. However, achieving sublinear convergence in this setting requires the following assumption, which can be viewed as a strengthening of (3.2) from Assumption 3.2.

Assumption 3.4. *There exists a pair $(M_3, M_4) \in (0,\infty) \times (0,\infty)$ (independent of k) such that, for all $k \in \mathbb{N}$, the squared norm of g_k satisfies*

$$\mathbb{E}_k[\|g_k\|^2] \leq M_3\alpha_k^2 + M_4\|\nabla f(x_k)\|^2. \tag{3.26}$$

One finds that Assumption 3.4 can be satisfied under reasonable conditions in practice if one employs mini-batch stochastic gradient estimates with sample sizes that increase with k; see, e.g., [9]. For example, in the context of problem (2.1), suppose that

$$g_k = \frac{1}{|\mathcal{S}_k|}\sum_{j \in \mathcal{S}_k} \nabla_x F(x_k, \xi_{k,j}), \tag{3.27}$$

where the values $\{\xi_{k,j}\}_{j \in \mathcal{S}_k}$ are drawn independently according to the distribution of ξ. If one assumes that the variance of each $\nabla_x F(x_k, \xi_{k,j})$ is equal and bounded by $M \in (0,\infty)$, then for arbitrary $j \in \mathcal{S}_k$ it follows (see, e.g., [8]) that

$$\mathbb{E}_k[\|g_k\|^2] - \|\nabla f(x_k)\|^2 \leq \frac{M}{|\mathcal{S}_k|}. \tag{3.28}$$

Hence, (3.26) holds with $M_3 = M$ and $M_4 = 1$ if one chooses $|\mathcal{S}_k| = \alpha_k^{-2}$. (In Theorem 3.3 below, the result requires $\alpha_k = \Theta(\frac{1}{k})$, in which case one can employ $|\mathcal{S}_k| = \Theta(k^2)$.)

An important consequence of Assumption 3.4 is the following, which strengthens Lemma 3.2.

Lemma 3.3. *Under Assumption 3.4, it follows that, for all $k \in \mathbb{N}$,*

$$\mathbb{P}_k[E_k]\mathbb{E}_k[\nabla f(x_k)^T g_k | E_k] \leq h_3\alpha_k + h_4\|\nabla f(x_k)\|^2 \tag{3.29}$$

for any $(h_3, h_4) \in (0,\infty) \times (0,\infty)$ such that $h_3 \geq \frac{1}{2}\sqrt{M_3}$ and $h_4 \geq \frac{1}{2}\sqrt{M_3}(\max_{k \in \mathbb{N}} \alpha_k) + \sqrt{M_4}$.

Proof. Proof. By Jensen's Inequality, concavity of the square root, and Assumption 3.4, one finds that

$$\mathbb{E}_k[\|g_k\|] \leq \sqrt{\mathbb{E}_k[\|g_k\|^2]} \leq \sqrt{M_3\alpha_k^2 + M_4\|\nabla f(x_k)\|^2} \leq \sqrt{M_3}\alpha_k + \sqrt{M_4}\|\nabla f(x_k)\|.$$

The result then follows using the same line of argument as used in the proof of Lemma 3.2. □

The following examples parallel Examples 3.1 and 3.2, but illustrate the attainment of (3.29).

Example 3.3. *Consider the scenario in Example 3.1. Then, if $\sigma_k \leq \alpha_k$ for all $k \in \mathbb{N}$ with $\alpha_k \leq \alpha$ for some $\alpha \in (0,\infty)$ for all $k \in \mathbb{N}$, it follows that (3.29) holds with*

$$h_3 = \frac{1}{2\sqrt{2\pi}} \quad \textit{and} \quad h_4 = 1 + \frac{\alpha}{2\sqrt{2\pi}}. \tag{3.30}$$

Example 3.4. *Consider the scenario in Example 3.2. Then, if $\sqrt{\mu_k^T\Sigma_k\mu_k} \leq \alpha_k$ for all $k \in \mathbb{N}$ with $\alpha_k \leq \alpha$ for some $\alpha \in (0,\infty)$ for all $k \in \mathbb{N}$, it follows that (3.29) holds with h_3 and h_4 from (3.30).*

Our next theorem on the behavior of TRish is now proved as the following. (For the result, we include Assumptions 3.2 and 3.4 for convenience since, in our proof, we employ results that we have proved using each of these assumptions. Notice, however, that the bound (3.2) in Assumption 3.2 holds under Assumption 3.4 if one considers $M_1 \geq M_3(\max_{k\in\mathbb{N}} \alpha_k^2)$ and $M_2 = M_4$.)

Theorem 3.3. *Under Assumptions 3.1, 3.2, 3.3, and 3.4, and with a pair (h_3, h_4) satisfying the inequalities in Lemma 3.3, suppose that TRish is run with $\gamma_1 > \gamma_2 > 0$ such that $\frac{\gamma_1}{\gamma_2} < \frac{h_4}{h_4-1}$ (meaning $\gamma_1 - h_4(\gamma_1-\gamma_2) > 0$), and with, for all $k \in \mathbb{N}$,*

$$\alpha_k = \frac{a}{b+k} \quad \textit{for some} \quad a \in \left(\frac{1}{2c\beta_1}, \frac{b+1}{2c\beta_1}\right) \quad \textit{and} \quad b > 0 \quad \textit{such that} \quad \alpha_1 \in \left(0, \frac{\gamma_1 - h_4(\gamma_1-\gamma_2)}{\gamma_1 L M_2}\right],$$

where

$$\beta_1 = \tfrac{1}{2}(\gamma_1 - h_4(\gamma_1-\gamma_2)) > 0. \tag{3.31}$$

Then, for all $k \in \mathbb{N}$, the expected optimality gap satisfies

$$\mathbb{E}[f(x_k)] - f_* \leq \frac{\nu}{b+k}, \tag{3.32}$$

where

$$\nu = \max\left\{\frac{a^2\beta_2}{2ac\beta_1 - 1}, (b+1)(f(x_1)-f_*)\right\} > 0 \tag{3.33}$$

$$\textit{and} \quad \beta_2 = h_3(\gamma_1-\gamma_2) + \tfrac{1}{2}\gamma_1^2 L M_1 > 0. \tag{3.34}$$

Proof. Proof. Similar to the proof of Theorem 3.1, for all $k \in \mathbb{N}$,

$$\begin{aligned}\mathbb{E}_k[f(x_{k+1})] - f(x_k) \leq & -\gamma_1\alpha_k(1 - \tfrac{1}{2}\gamma_1 L M_2 \alpha_k)\|\nabla f(x_k)\|^2 \\ & + (\gamma_1-\gamma_2)\alpha_k(h_3\alpha_k + h_4\|\nabla f(x_k)\|^2) + \tfrac{1}{2}\gamma_1^2 L M_1 \alpha_k^2.\end{aligned}$$

Therefore, with (β_1, β_2) defined in (3.31)/(3.34), it follows with (3.11) that, for all $k \in \mathbb{N}$,

$$\begin{aligned}\mathbb{E}_k[f(x_{k+1})] - f(x_k) &\leq -\beta_1\alpha_k\|\nabla f(x_k)\|^2 + \beta_2\alpha_k^2 \\ &\leq -2c\beta_1\alpha_k(f(x_k)-f_*) + \beta_2\alpha_k^2.\end{aligned} \tag{3.35}$$

Adding and subtracting f_* on the left-hand side, taking total expectations, and rearranging yields

$$\mathbb{E}[f(x_{k+1})] - f_* \leq (1 - 2c\beta_1\alpha_k)(\mathbb{E}[f(x_k)] - f_*) + \beta_2\alpha_k^2.$$

Using this inequality, which has the same form as (3.25), one can apply the same inductive argument as in the remainder of the proof of Theorem 3.2 to achieve the desired result. □

Overall, we have proved two theorems for TRish when diminishing stepsizes are employed. If the sequence $\{\gamma_{k,1} - \gamma_{k,2}\}$ diminishes proportionally with $\{\alpha_k\}$, then sublinear convergence of the expected optimality gap is achieved under the same assumptions as needed for such a result for an SG method. We followed this with a result for the case when $\{\gamma_{k,1} - \gamma_{k,2}\}$ is constant, in which case a sublinear convergence result for the expected optimality gap requires that the stochastic gradient estimates satisfy Assumption 3.4.

3.3 P-L Condition, Constant Parameters, and Linearly Decreasing Variance

Let us now prove a convergence result for TRish when the P-L condition holds, each sequence $\{\alpha_k\}$, $\{\gamma_{1,k}\}$, and $\{\gamma_{2,k}\}$ is constant, and the stochastic gradients satisfy the following assumption.

Assumption 3.5. *There exist constants* $(M_5, \zeta) \in (0,\infty) \times (0,1)$ *such that*

$$\mathbb{E}_k[\|g_k\|^2] \leq M_5 \zeta^{k-1} + \|\nabla f(x_k)\|^2. \tag{3.36}$$

The achievement of linear convergence of the expected optimality gap for SG also requires increasingly accurate gradient estimates along the lines required in Assumption 3.5; see, e.g., [2]. One finds that Assumption 3.5 can be satisfied under reasonable conditions in practice if one employs mini-batch stochastic gradient estimates with sample sizes that increase with k. For example, using estimates as in (3.27) and under the same conditions as led to (3.28), one finds that (3.36) holds if the sample sizes increase geometrically, e.g., $|\mathcal{S}_k| = \lceil \tau^{k-1} \rceil$ for some $\tau \in (1,\infty)$.

Our main result in this section, namely, Theorem 3.4, requires the following.

Lemma 3.4. *Under the Assumption 3.5, it follows that, for all* $k \in \mathbb{N}$,

$$\mathbb{P}_k[E_k]\mathbb{E}_k[\nabla f(x_k)^T g_k | E_k] \leq h_5 \lambda^{k-1} + h_6 \|\nabla f(x_k)\|^2 \tag{3.37}$$

for any $(h_5, h_6) \in (0,\infty) \times (0,\infty) \times (0,1)$ *such that* $h_5 \geq \frac{1}{2}\sqrt{M_5}$, $1 + h_6 \geq \frac{1}{2}\sqrt{M_5}$, *and* $\lambda \geq \sqrt{\zeta}$.

Proof. Proof. By Jensen's inequality, concavity of the square root, and Assumption 3.5, one finds that

$$\mathbb{E}_k[\|g_k\|] \leq \sqrt{\mathbb{E}_k[\|g_k\|^2]} \leq \sqrt{M_5 \zeta^{k-1} + \|\nabla f(x_k)\|^2} \leq \sqrt{M_5}(\sqrt{\zeta})^{k-1} + \|\nabla f(x_k)\|. \tag{3.38}$$

The result then follows using the same line of argument as used in the proof of Lemma 3.2. □

The following examples parallel Examples 3.1 and 3.2, but illustrate the attainment of (3.37).

Example 3.5. *Consider the scenario in Example 3.1. Then, since (3.36) implies that* $\sigma_k^2 \leq M_3 \zeta^{k-1}$ *for all* $k \in \mathbb{N}$, *it follows along with the fact that* $\zeta \in (0,1)$ *that*

$$\begin{aligned}\mathbb{P}_k[E_k]\mathbb{E}_k[\nabla f(x_k)^T g_k | E_k] &\leq \frac{\sigma_k}{2\sqrt{2\pi}} + \left(1 + \frac{\sigma_k}{2\sqrt{2\pi}}\right)\mu_k^2 \\ &\leq \frac{\sqrt{M_3}}{2\sqrt{2\pi}}(\sqrt{\zeta})^{k-1} + \left(1 + \frac{\sqrt{M_3}}{2\sqrt{2\pi}}\right)\mu_k^2.\end{aligned}$$

Hence, it follows that (3.37) holds with

$$h_5 = \frac{\sqrt{M_3}}{2\sqrt{2\pi}}, \quad h_6 = 1 + \frac{\sqrt{M_3}}{2\sqrt{2\pi}}, \quad \text{and} \quad \lambda = \sqrt{\zeta}. \tag{3.39}$$

Example 3.6. *Consider the scenario in Example 3.2. Then, with* $\sqrt{\mu_k^T \Sigma_k \mu_k} \leq M_3 \zeta^{k-1}$ *for all* $k \in \mathbb{N}$, *it follows that (3.37) holds with* h_5, h_6, *and* λ *from (3.39).*

Our next theorem on the behavior of TRish is now proved as the following. (For the result, we include Assumptions 3.2 and 3.5 for convenience since, in our proof, we employ results that we have proved using each of these assumptions. Notice, however, that the bound (3.2) in Assumption 3.2 holds under Assumption 3.5 if one considers $M_1 \geq M_5$, $M_2 \geq 1$, and any $\zeta \in (0,1)$.)

Theorem 3.4. *Under Assumptions 3.1, 3.2, 3.3, and 3.5, and with a tuple (h_5, h_6, λ) satisfying the inequalities in Lemma 3.4, suppose that TRish is run with $\gamma_1 > \gamma_2 > 0$ such that $\frac{\gamma_1}{\gamma_2} < \frac{h_6}{h_6-1}$ (meaning $\gamma_1 - h_6(\gamma_1 - \gamma_2) > 0$), and with $\alpha_k = \alpha$ for all $k \in \mathbb{N}$ such that*

$$0 < \alpha \leq \min\left\{\frac{\gamma_1 - h_6(\gamma_1-\gamma_2)}{\gamma_1^2 L}, \frac{1}{c\kappa_1}\right\}, \tag{3.40}$$

where

$$\kappa_1 := \tfrac{1}{2}(\gamma_1 - h_6(\gamma_1 - \gamma_2)) > 0. \tag{3.41}$$

Then, for all $k \in \mathbb{N}$, the expected optimality gap satisfies

$$\mathbb{E}[f(x_k)] - f_* \leq \omega\rho^{k-1}, \tag{3.42}$$

where

$$\begin{aligned} \kappa_2 &:= h_5(\gamma_1-\gamma_2) + \tfrac{1}{2}\gamma_1^2 \alpha L M_3 > 0, \\ \omega &:= \max\{f(x_1) - f_*, \tfrac{\kappa_2}{c\kappa_1}\} > 0, \\ \textit{and}\quad \rho &:= \max\{1 - \alpha c\kappa_1, \lambda, \zeta\} \in (0,1). \end{aligned} \tag{3.43}$$

Proof. Proof. As in the proof of Lemma 3.1, it follows with (3.36) and (3.37) that, for all $k \in \mathbb{N}$,

$$\begin{aligned} &\mathbb{E}_k[f(x_{k+1})] - f(x_k) \\ \leq\ & -\alpha\gamma_1\|\nabla f(x_k)\|^2 + (\gamma_1-\gamma_2)\alpha\mathbb{P}_k[E_k]\mathbb{E}_k[\nabla f(x_k)^T g_k | E_k] + \tfrac{1}{2}\gamma_1^2 L\alpha^2 \mathbb{E}_k[\|g_k\|^2] \\ \leq\ & -\alpha\gamma_1\|\nabla f(x_k)\|^2 + (\gamma_1-\gamma_2)\alpha(h_5\lambda^{k-1} + h_6\|\nabla f(x_k)\|^2) + \tfrac{1}{2}\gamma_1^2 L\alpha^2(M_3\zeta^{k-1} + \|\nabla f(x_k)\|^2) \\ =\ & -\alpha(\gamma_1 - h_6(\gamma_1-\gamma_2) - \tfrac{1}{2}\gamma_1^2 L\alpha)\|\nabla f(x_k)\|^2 + (\gamma_1-\gamma_2)\alpha h_5\lambda^{k-1} + \tfrac{1}{2}\gamma_1^2 L\alpha^2 M_3\zeta^{k-1} \\ \leq\ & -\tfrac{1}{2}\alpha(\gamma_1 - h_6(\gamma_1-\gamma_2))\|\nabla f(x_k)\|^2 + (\gamma_1-\gamma_2)\alpha h_5\lambda^{k-1} + \tfrac{1}{2}\gamma_1^2 L\alpha^2 M_3\zeta^{k-1}. \end{aligned}$$

Therefore, with (κ_1, κ_2) defined in (3.41)/(3.43), it follows with (3.11) that, for all $k \in \mathbb{N}$,

$$\begin{aligned} \mathbb{E}_k[f(x_{k+1})] &\leq f(x_k) - \alpha\kappa_1\|\nabla f(x_k)\|^2 + \alpha\kappa_2 \max\{\lambda,\zeta\}^{k-1} \\ &\leq f(x_k) - 2\alpha c\kappa_1(f(x_k) - f_*) + \alpha\kappa_2\max\{\lambda,\zeta\}^{k-1}, \end{aligned}$$

from which it follows that

$$\mathbb{E}[f(x_{k+1})] - f_* \leq (1 - 2\alpha c\kappa_1)(\mathbb{E}[f(x_k)] - f_*) + \alpha\kappa_2\max\{\lambda,\zeta\}^{k-1}.$$

Let us now prove (3.42) by induction. First, for $k=1$, the inequality follows by the definition of ω. Then, assuming the inequality holds true for $k \in \mathbb{N}$, one finds that

$$\begin{aligned} \mathbb{E}[f(x_{k+1})] - f_* &\leq (1 - 2\alpha c\kappa_1)(\mathbb{E}[f(x_k)] - f_*) + \alpha\kappa_2\max\{\lambda,\zeta\}^{k-1} \\ &\leq (1-2\alpha c\kappa_1)\omega\rho^{k-1} + \alpha\kappa_2\max\{\lambda,\zeta\}^{k-1} \\ &= \omega\rho^{k-1}\left(1 - 2\alpha c\kappa_1 + \frac{\alpha\kappa_2}{\omega}\left(\frac{\max\{\lambda,\zeta\}}{\rho}\right)^{k-1}\right) \\ &\leq \omega\rho^{k-1}\left(1 - 2\alpha c\kappa_1 + \frac{\alpha\kappa_2}{\omega}\right) \\ &\leq \omega\rho^{k-1}(1 - \alpha c\kappa_1) \\ &\leq \omega\rho^k, \end{aligned}$$

which proves that the conclusion holds for $k+1$, as desired. □

3.4 No P-L Condition and Constant Parameters

Let us now consider the behavior of TRish when the P-L condition does not hold. Our first such result involves the use of constant $\{\gamma_{1,k}\}$, $\{\gamma_{2,k}\}$, and $\{\alpha_k\}$.

Theorem 3.5. *Under Assumptions 3.1 and 3.2 and with a pair (h_1, h_2) satisfying the inequalities in Lemma 3.2, suppose that TRish is run with $(\gamma_{1,k}, \gamma_{2,k}) = (\gamma_1, \gamma_2)$ for all $k \in \mathbb{N}$ such that $\frac{\gamma_1}{\gamma_2} < \frac{h_2}{h_2-1}$ (meaning $\gamma_1 - h_2(\gamma_1 - \gamma_2) > 0$) and with $\alpha_k = \alpha$ for all $k \in \mathbb{N}$ such that*

$$0 < \alpha \leq \frac{\gamma_1 - h_2(\gamma_1 - \gamma_2)}{\gamma_1 L M_2}.$$

Then, with (θ_1, θ_2) defined in (3.15)/(3.17), it follows that, for all $K \in \mathbb{N}$,

$$\mathbb{E}\left[\sum_{k=1}^{K} \|\nabla f(x_k)\|^2\right] \leq \frac{K\theta_2}{\alpha\theta_1} + \frac{f(x_1) - f_*}{\alpha\theta_1} \tag{3.44a}$$

$$\textit{and}\ \ \mathbb{E}\left[\frac{1}{K}\sum_{k=1}^{K} \|\nabla f(x_k)\|^2\right] \leq \frac{\theta_2}{\alpha\theta_1} + \frac{f(x_1) - f_*}{K\alpha\theta_2} \xrightarrow{K\to\infty} \frac{\theta_2}{\alpha\theta_1}. \tag{3.44b}$$

Proof. Proof. As in the proof of Theorem 3.1, combining the results of Lemmas 3.1 and 3.2, it follows that the inequality (3.18) holds for all $k \in \mathbb{N}$. Taking total expectations, it follows that, for all $k \in \mathbb{N}$,

$$\mathbb{E}[f(x_{k+1})] - \mathbb{E}[f(x_k)] \leq -\alpha\theta_1 \mathbb{E}[\|\nabla f(x_k)\|^2] + \theta_2.$$

Summing both sides for $k \in \{1, \dots, K\}$ yields

$$f_* - f(x_1) \leq \mathbb{E}[f(x_{K+1})] - f(x_1) \leq -\alpha\theta_1 \sum_{k=1}^{K} \mathbb{E}[\|\nabla f(x_k)\|^2] + K\theta_2.$$

Rearranging yields (3.44a), then dividing by K yields (3.44b). □

As in the case of [2, Thm. 4.8], this result shows that while one cannot bound the expected optimality gap as when the P-L condition holds, one can bound the average norm of the gradients of the objective that are observed during the optimization process.

3.5 No P-L Condition and Sublinearly Diminishing Stepsizes

Finally, let us consider the behavior of TRish when the P-L condition does not hold and diminishing stepsizes are employed. For brevity, the following theorem considers both when parameters are chosen as in Theorem 3.2 and as in Theorem 3.3, since in either case the final conclusion is the same.

Theorem 3.6. *Suppose Assumptions 3.1 and 3.2 hold and at least one of the following.*

(i) With a pair (h_1, h_2) satisfying the inequalities in Lemma 3.2, suppose that TRish is run with $\{\gamma_{1,k}\}$, $\{\gamma_{2,k}\}$, and $\{\alpha_k\}$ chosen as in Theorem 3.2.

(ii) Suppose Assumption 3.4 holds and, with a pair (h_3, h_4) satisfying the inequalities in Lemma 3.3, suppose that TRish is run with $\{\gamma_{1,k}\}$, $\{\gamma_{2,k}\}$, and $\{\alpha_k\}$ chosen as in Theorem 3.3.

Then, with $A_K := \sum_{k=1}^{K} \alpha_k$, it follows that

$$\lim_{K\to\infty} \mathbb{E}\left[\sum_{k=1}^{K} \alpha_k \|\nabla f(x_k)\|^2\right] < \infty \tag{3.45a}$$

$$\textit{and}\ \ \mathbb{E}\left[\frac{1}{A_K}\sum_{k=1}^{K} \alpha_k \|\nabla f(x_k)\|^2\right] \xrightarrow{K\to\infty} 0. \tag{3.45b}$$

Proof. Proof. First observe that, under the conditions of the theorem, specifically the conditions placed on the stepsize sequence $\{\alpha_k\}$ in Theorem 3.2 or Theorem 3.3, it follows that

$$\sum_{k=1}^{\infty} \alpha_k = \infty \quad \text{and} \quad \sum_{k=1}^{\infty} \alpha_k^2 < \infty. \tag{3.46}$$

Second, following the proofs of Theorem 3.2 or Theorem 3.3, it follows that under conditions (i) or (ii) one finds by taking total expectations in (3.24) or (3.35) that

$$\begin{aligned} &\mathbb{E}[f(x_{k+1})] - \mathbb{E}[f(x_k)] \leq -\tfrac{1}{2}\gamma_1 \alpha_k \mathbb{E}[\|\nabla f(x_k)\|^2] + \delta \alpha_k^2 \\ \text{or} \quad &\mathbb{E}[f(x_{k+1})] - \mathbb{E}[f(x_k)] \leq -\beta_1 \alpha_k \mathbb{E}[\|\nabla f(x_k)\|^2] + \beta_2 \alpha_k^2. \end{aligned}$$

Without loss of generality, let us assume that condition (ii) holds and the latter inequality above is satisfied. (The proof is the same if condition (i) holds and the former inequality above is satisfied.) Summing both sides for $k \in \{1, \dots, K\}$ yields

$$f_* - f(x_1) \leq \mathbb{E}[f(x_{K+1})] - f(x_1) \leq -\beta_1 \sum_{k=1}^{K} \alpha_k \mathbb{E}[\|\nabla f(x_k)\|^2] + \beta_2 \sum_{k=1}^{K} \alpha_k^2,$$

which after rearrangement gives

$$\sum_{k=1}^{K} \alpha_k \mathbb{E}[\|\nabla f(x_k)\|^2] \leq \frac{f(x_1) - f_*}{\beta_1} + \frac{\beta_2}{\beta_1} \sum_{k=1}^{K} \alpha_k^2.$$

From (3.46), it follows that the right-hand side converges to a finite limit as $K \to \infty$, giving (3.45a). Then, the limit (3.45b) follows since (3.46) ensures that $\{A_K\} \to \infty$ as $K \to \infty$. □

A consequence of this theorem is the straightforward fact that

$$\liminf_{k \to \infty} \mathbb{E}[\|\nabla f(x_k)\|^2] = 0.$$

That is, under the conditions of the theorem, the expected squared norms of the gradients at the iterates of the algorithm cannot stay bounded away from zero.

4 Numerical Experiments

In this section, we provide the results of numerical experiments to demonstrate the performance of TRish compared to a stochastic gradient (SG) approach. Through solving machine learning test problems involving objective functions of the form (2.2)—some convex and some nonconvex—we demonstrate that TRish can outperform SG with comparable computational effort. Before presenting our results, we first discuss how the parameters of the algorithm might be chosen.

4.1 Algorithm Parameter Selection

Our analysis in §3 provides guidelines on how the stepsizes $\{\alpha_k\}$ and pairs $\{(\gamma_{1,k}, \gamma_{2,k})\}$ should be chosen to guarantee convergence properties for TRish. That said, as for SG, the values required by the theory are often too conservative in practice, whereas one often finds better performance by a parameter tuning scheme. Still, it is worthwhile to comment on how the theoretical analysis might inform parameter selection. For our purposes, since our numerical experiments focus on results obtained with fixed parameters, we shall discuss how the analysis in §3.1 informs parameter selection. Similar conclusions can be drawn based on our other theoretical results.

For simplicity, let us assume that the bound (3.2) in Assumption 3.2 holds with $M_2 = 1$. In this case, the bound (3.2) is equivalent to the restriction that the variance of the stochastic gradient estimate is bounded by M_1, i.e., that $\mathbb{E}_k[\|g_k\|^2] - \|\nabla f(x_k)\|^2 \leq M_1$. If one has an estimate $\widetilde{M}_1$ of M_1—which, for example, can be obtained by sampling gradients and computing a variance estimate—then, following Lemma 3.2, one can employ the value $\widetilde{h}_2 = \frac{1}{2}\sqrt{\widetilde{M}_1} + 1$ for parameter selection. In particular, Theorem 3.1 suggests to choose (γ_1, γ_2) such that

$$\frac{\gamma_1}{\gamma_2} < \frac{\widetilde{h}_2}{\widetilde{h}_2 - 1} = 1 + \frac{2}{\sqrt{\widetilde{M}_1}}.$$

Naturally, this still leads to flexibility in the precise values of (γ_1, γ_2), but the trade-offs between different choices become similar to the traditional trade-offs one finds for the selection of α in an SG scheme: (i) one can choose values such that $\gamma_1 - \gamma_2$ is large, which leads to fast convergence, but only to a relatively large neighborhood of the solution, or (ii) one can choose values such that $\gamma_1 - \gamma_2$ is small, which leads to slow convergence, but to a relatively small neighborhood of a solution. Overall, one might be discouraged by the idea that the choice of (γ_1, γ_2) requires estimation of the upper bound M_1. However, this is not dissimilar to the fact that, theoretically, one needs an estimate of the Lipschitz constant L of the gradient in order to choose the stepsize for SG, and clearly also for TRish, such as through the bound (3.14). The good news is that estimating the variance of the stochastic gradient estimates is a reasonable request that could even be done during an initial phase that simply uses SG iterations.

Despite all of this commentary, in practice one should expect to achieve better performance by simply tuning parameters for a given problem, as is often done for SG methods. For our experiments described in the following subsections, we chose (γ_1, γ_2) by a simple tuning scheme that also selects the stepsize α. We took care to make sure that the tuning procedure for TRish did not require more effort than the tuning used for the SG method that we have for comparison purposes.

4.2 Logistic Regression

As a first test case, we considered the problem of binary classification through logistic regression using a few datasets available in the well-known LIBSVM repository; see [4]. In particular, for each dataset, with training feature vector $z_i \in \mathbb{R}^n$ and training label $y_i \in \{-1, 1\}$ for all $i \in \{1, \dots, N\}$, the objective of this problem has the form

$$f(x) = \frac{1}{N}\sum_{i=1}^{N} \log(1 + e^{-y_i(x^T z_i)}). \tag{4.1}$$

Also available in each case is a testing dataset $\{(\bar{z}_i, \bar{y}_i)\}_{i=1}^{\overline{N}}$.

We ran implementations of TRish and SG and compare performance by comparing *training loss* (i.e., the objective function (4.1) evaluated with the training data) and *testing accuracy* (i.e., for a given approximate solution, what fraction of the testing set is classified correctly) for iterates throughout the optimization process. We ran each algorithm for one epoch (i.e., until N training pairs have been accessed) with a fixed stepsize α and, for TRish, a fixed parameter pair (γ_1, γ_2).

For both algorithms and all datasets, the stochastic gradient estimates were computed using a mini-batch size of 64. For choosing a fair set of parameters for the comparison for each dataset, we first ran SG with a stepsize of 0.1 and computed G as the average norm of stochastic gradient estimates throughout the run. Then, for TRish, we considered the stepsizes $\alpha \in \{10^{-1}, 10^{-1/2}, 10^0, 10^{1/2}, 10^1\}$ and parameters $\gamma_1 \in \{\frac{4}{G}, \frac{8}{G}, \frac{16}{G}, \frac{32}{G}\}$ and $\gamma_2 \in \{\frac{1}{2G}, \frac{1}{G}, \frac{2}{G}\}$. (The value G gauges the magnitude of the stochastic gradient estimates, which depends on problem scaling. As seen in our results, these choices of (γ_1, γ_2) ensure that step normalization—i.e., case 2 of TRish—occurs. In practice, one could compute G during an initial SG phase before starting TRish, but to cleanly distinguish between TRish and SG, we computed this value using an independent run of SG.) This resulted in 60 parameter settings with TRish employing stepsizes in the range from $\frac{1}{2G} \times 10^{-1}$ (i.e., the minimum γ_2 times the minimum α) to $\frac{32}{G} \times 10^1$ (i.e., the maximum γ_1 times the maximum α). Hence, for SG, we considered 60 values for α in the range $[\frac{1}{2G} \times 10^{-1}, \frac{32}{G} \times 10^1]$ so that neither algorithm had an advantage in terms of the range of the stepsizes. Specifically, we considered the 60 values such that $\log_{10}(\alpha)$ was evenly distributed in $[\log_{10}(\frac{1}{2G} \times 10^{-1}), \log_{10}(\frac{32}{G} \times 10^{1})]$.

For each dataset, we ran the algorithms with these different parameters settings and selected for each the setting that led to the best average testing accuracy in the last ten iterations of the run.

4.2.1 a1a.

The first dataset that we considered was `a1a` in which the feature vectors have length $n = 123$, the number of points in the training set is $N = 1605$, and the number of points in the testing set is $\overline{N} = 30956$. For tuning, the value $G \approx 0.1746$ was determined, yielding a stepsize range of approximately $[0.2863, 1832]$. After tuning, the selected parameter setting for TRish was $(\alpha, \gamma_1, \gamma_2) \approx (0.1, 22.90, 2.863)$ and the selected parameter setting for SG was $\alpha \approx 0.4471$.

The algorithms, TRish and SG, were each run 10 times from the same starting point (the origin). The training losses and testing accuracies, averaged over these 10 runs, are plotted in Figure 2 after 0.1 epoch through the end of the first epoch. (The values during the first 0.1 epoch are not plotted here, nor for the other datasets, so that it is easier to distinguish the differences at the end of the first epoch.) It is worthwhile to note that, during the runs for TRish, case 1 did not occur, case 2 occurred in approximately 99% of the iterations, and case 3 occurred in approximately 1% of the iterations; i.e., step normalization occurred in a large majority of the iterations. The figure shows that TRish yields better training losses throughout the optimization process. However, for this dataset, the performance in terms of testing accuracy is roughly the same for both algorithms.

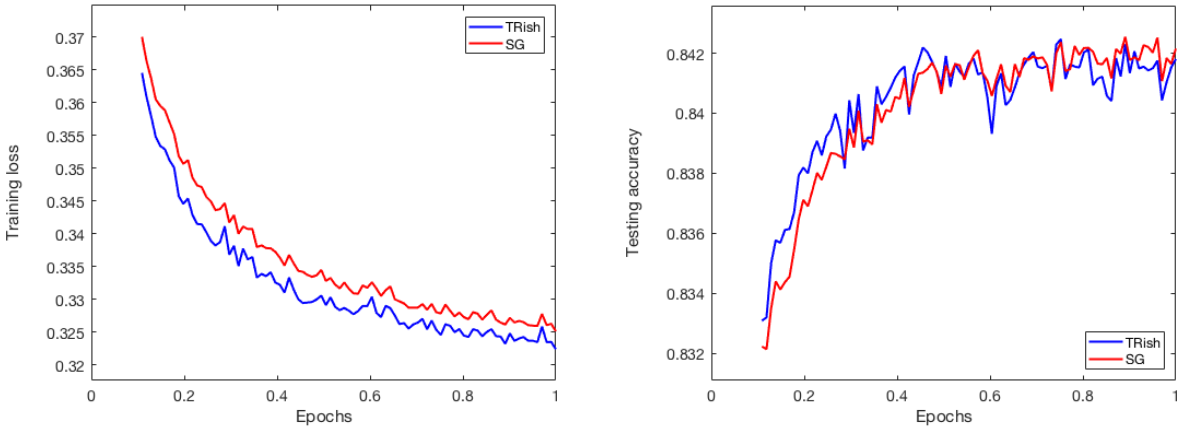

Figure 2: Average training loss and testing accuracy during the first epoch when TRish and SG are employed to minimize the logistic regression function (4.1) using the `a1a` dataset.

4.2.2 w1a.

The second dataset that we considered was `w1a` in which the feature vectors have length $n = 300$, the number of points in the training set is $N = 2477$, and the number of points in the testing set is $\overline{N} = 47272$. For tuning, the value $G \approx 0.0887$ was determined, yielding a stepsize range of approximately $[0.5638, 3608]$. After tuning, the selected parameter setting for TRish was $(\alpha, \gamma_1, \gamma_2) \approx (0.1, 360.8, 5.638)$ and the selected parameter setting for SG was $\alpha \approx 0.6541$.

The training losses and testing accuracies, averaged over 10 runs when both algorithms were initialized at the same starting point (the origin), are plotted in Figure 3. During the runs for TRish, case 2 occurred in approximately 99% of the iterations while case 1 and case 3 combined occurred in fewer than 1% of the iterations. For this dataset, TRish outperformed SG both in terms of training losses and testing accuracies throughout the first epoch.

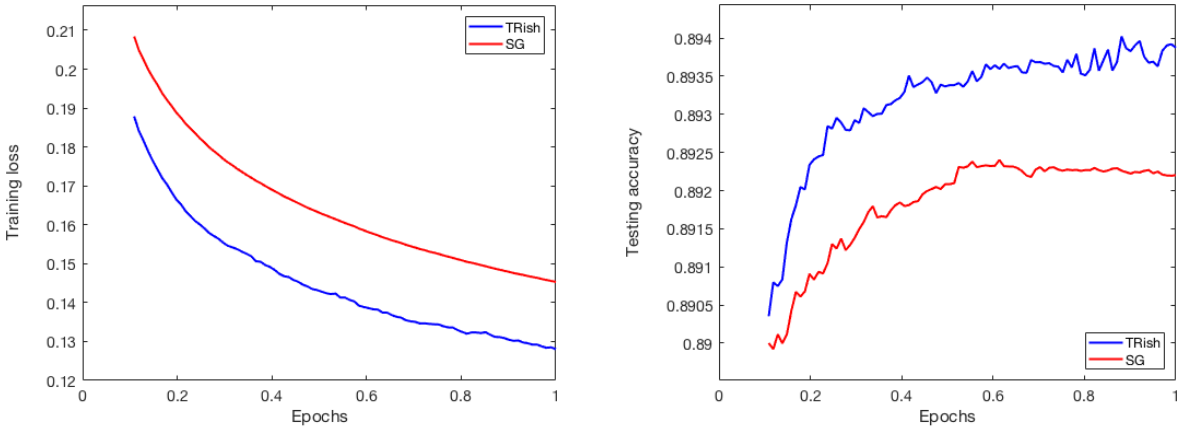

Figure 3: Average training loss and testing accuracy during the first epoch when TRish and SG are employed to minimize the logistic regression function (4.1) using the `w1a` dataset.

4.2.3 rcv1.

The third dataset that we considered was `rcv1` in which the feature vectors have length $n = 47236$, the number of points in the training set is $N = 20242$, and the number of points in the testing set is $\overline{N} = 677399$. For tuning, the value $G \approx 0.0497$ was determined, yielding a stepsize range of approximately $[1.007, 6444]$. After tuning, the selected parameter setting for TRish was $(\alpha, \gamma_1, \gamma_2) \approx (0.3162, 644.4, 10.07)$ and the selected parameter setting for SG was $\alpha \approx 10.84$.

The training losses and testing accuracies, averaged over 10 runs when both algorithms were initialized at the same starting point (the origin), are plotted in Figure 4. During the runs for TRish, case 1 occurred in approximately 27% of the iterations, case 2 occurred in approximately 73% of the iterations, and case 3 did not occur. For this dataset, TRish outperformed SG both in terms of training losses and testing accuracies throughout the first epoch. That said, the testing accuracies appear to near at the end of the first epoch, leading one to wonder about the performance of the methods if the parameters are re-tuned and the algorithms are run for more epochs.

To address this question, Figure 5 plots the training losses and testing accuracies—averaged over 10 runs—for TRish and SG during two epochs. (For this horizon, tuning led to the parameter setting for TRish as $(\alpha, \gamma_1, \gamma_2) = (0.1, 376.2, 47.02)$ and the parameter setting for SG as $\alpha \approx 5.192$. For TRish, case 2 occurred in approximately 94% of the iterations, case 3 occurred in approximately 5% of the iterations, and case 1 occurred in fewer than 1% of the iterations.) These plots show a trade-off where, for a longer horizon, the better parameters for TRish do not necessarily offer better results initially, but do offer better results eventually.

In all of the experiments presented in this section, TRish generally outperforms SG. However, the gains are somewhat limited due to the fact that, by convexity of the problems, both algorithms are tending to neighborhoods around the same optimal solution. The results presented in the next subsection, in which we consider nonconvex optimization problems arising from neural network training, show more substantial benefits from using TRish as compared to SG.

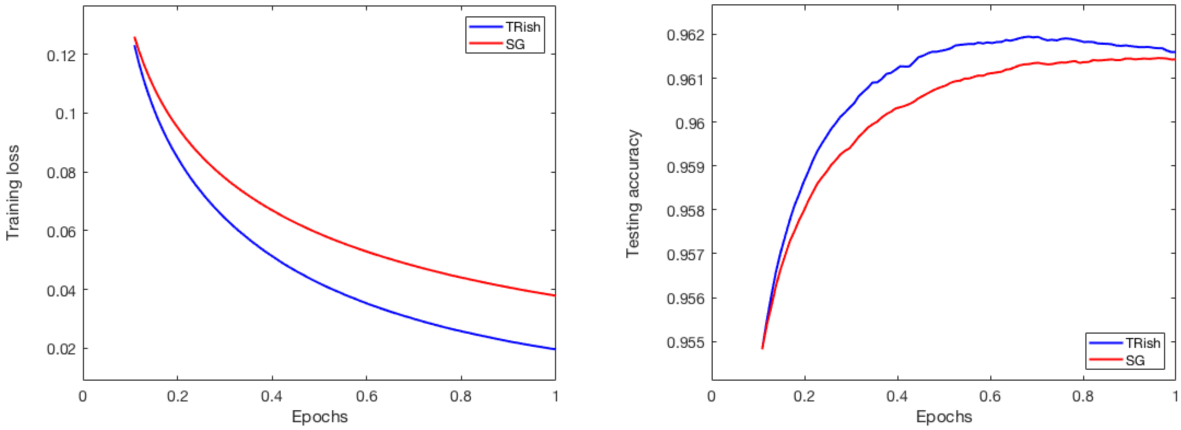

Figure 4: Average training loss and testing accuracy during the first epoch when TRish and SG are employed to minimize the logistic regression function (4.1) using the `rcv1` dataset.

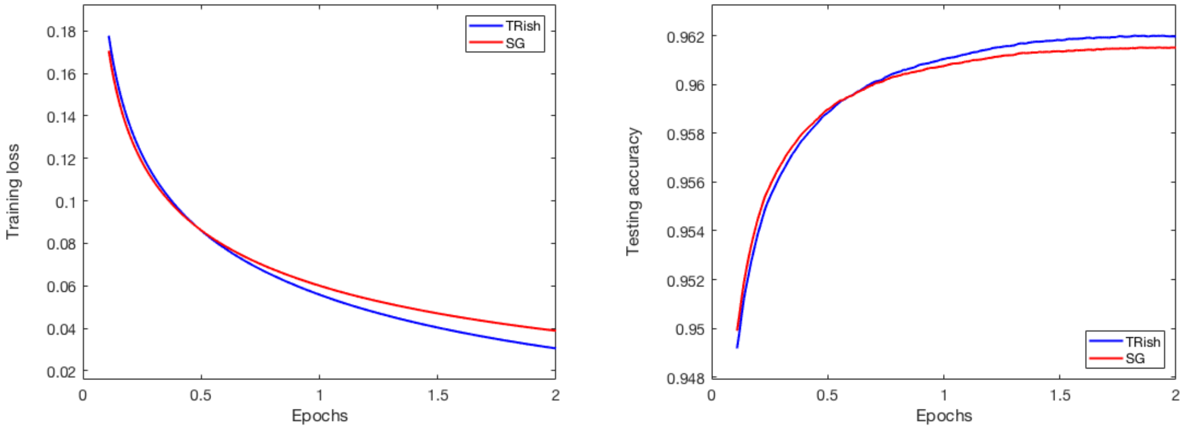

Figure 5: Average training loss and testing accuracy during the first two epochs when TRish and SG are employed to minimize the logistic regression function (4.1) using the `rcv1` dataset.

4.3 Neural Network Training

As a second test case, we considered the problem to train convolutional neural networks (CNNs) for image classification. We considered two well-known datasets. The first, the `mnist` dataset [17], is a collection of images of hand-written digits. The goal for training the network for this dataset is to classify which of the digits (0 through 9) is written in each image. It includes $N = 60000$ training samples and $\overline{N} = 10000$ testing samples. The second, the `cifar-10` dataset [15], is a collection of color images in ten categories (e.g., airplanes, dogs, and ships). The goal for training the network for this dataset is to classify the image with the correct category. It includes $N = 50000$ training samples and $\overline{N} = 10000$ testing samples.

Implemented using `tensorflow`, the neural networks that we considered for both datasets are composed of two convolutional layers (involving 32 and 64 filters, respectively, and each followed by an average pooling layer) followed by two fully connected layers. ReLU activation is used at each hidden layer and the objective is defined using the logistic (cross entropy) loss function. The networks vary slightly, e.g., due to the fact that a pixel for each `mnist` image corresponds to a single feature while a pixel for each `cifar-10` image corresponds to three features (for each RGB value since they are color images). As seen in our experimental results, training the network led to a very good classifier for `mnist`, yielding over 95% testing accuracy. The performance is less impressive for `cifar-10` (yielding around 60% accuracy); achieving higher accuracy would require a more sophisticated network and more computational resources than were available. That said, both datasets provide interesting settings for comparing the performance of TRish and SG.

As for the results in §4.2, we compare performance between TRish and SG by comparing training loss and testing accuracy. We tuned parameters using the same setup as in §4.2, except with slightly different parameter choices. In particular, the mini-batch size used when computing stochastic gradients was 128 and, when computing G, we ran SG with a stepsize of 0.01. For TRish, we considered stepsizes $\alpha \in \{10^{-3}, 10^{-2}, 10^{-1}, 10^{0}\}$ and parameters $\gamma_1 \in \{\frac{4}{G}, \frac{8}{G}, \frac{16}{G}\}$ and $\gamma_2 \in \{\frac{1}{8G}, \frac{1}{4G}, \frac{1}{2G}\}$. This means that SG was tuned with 36 choices of α in the range $[\frac{1}{8G} \times 10^{-3}, \frac{16}{G} \times 10^{0}]$.

4.3.1 `mnist`.

For `mnist`, we ran the algorithms for two epochs. For parameter tuning, the value $G \approx 2.8277$ was determined, yielding a stepsize range of approximately $[2.683 \times 10^{-5}, 3.435]$. After tuning, the selected parameter setting for TRish was $(\alpha, \gamma_1, \gamma_2) \approx (1, 1.717, 0.0268)$ and the selected parameter setting for SG was $\alpha \approx 0.0609$.

The training losses and testing accuracies for each of the 10 runs that we performed with the tuned parameters are plotted in Figure 6, ignoring the first 0.2 epochs so that the later values are more easily distinguished. (For each run, the network parameters were initialized to the same randomly generated values; the values were generated from a truncated normal distribution with mean 0 and standard deviation 0.1. We did not average the loss and accuracy values over the 10 runs since the optimization problem is nonconvex, meaning that for each run an algorithm might tend toward a different region of the search space.) During the runs for TRish, case 1 occurred in approximately 62% of the iterations, case 2 occurred in approximately 37% of the iterations, and case 3 almost did not occur. Overall, TRish consistently outperformed SG in terms of both training loss and testing accuracy throughout the optimization process.

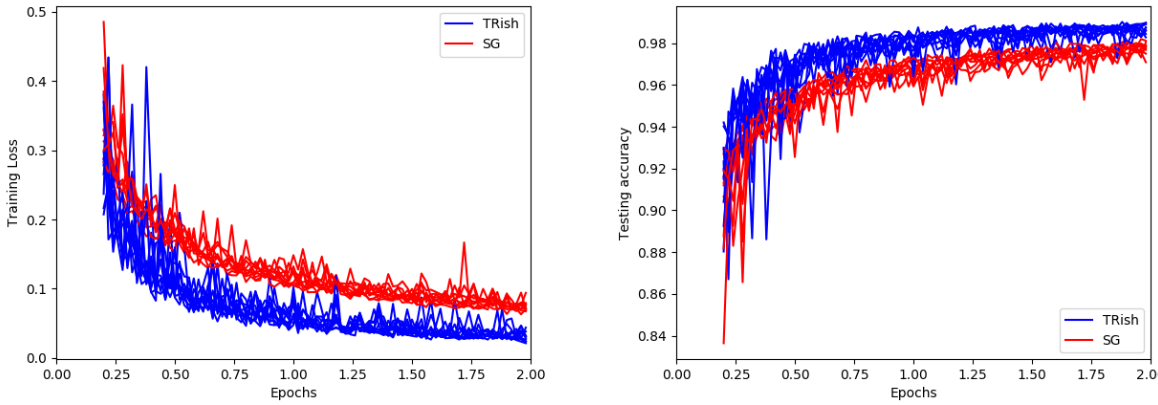

Figure 6: Average training loss and testing accuracy during the first two epochs when TRish and SG are employed to train a convolutional neural network using the `mnist` dataset.

4.3.2 `cifar-10`.

For `cifar-10`, we ran the algorithms for five epochs (since further improvement was clearly being made even after the first few epochs). The value $G \approx 964.39$ was determined, yielding a stepsize range of approximately $[8.990 \times 10^{-6}, 1.151]$. After tuning, the parameter setting for TRish was $(\alpha, \gamma_1, \gamma_2) \approx (1, 1.051, 0.0089)$ and the parameter setting for SG was $\alpha \approx 0.0104$.

The training losses and testing accuracies for each of the 10 runs that we performed with the tuned parameters are plotted in Figure 7, again ignoring the first 10% of the runs (i.e., in this case, the first 0.5 epochs) so that the later values are more easily distinguished. (For each run, the network parameters were initialized to the same randomly generated values; the values were generated from a truncated normal distribution with mean 0 and standard deviation 0.01.) During the runs for TRish, case 1 occurred in approximately 1% of the iterations, case 2 occurred in approximately 99% of the iterations, and case 3 did not occur. In these experiments, TRish typically outperformed SG in terms of both training loss and testing accuracy throughout each run.

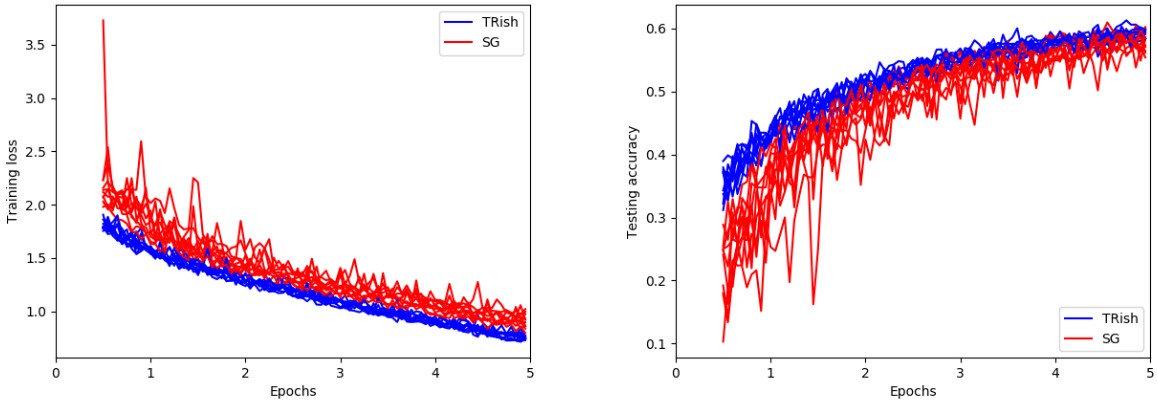

Figure 7: Average training loss and testing accuracy during the first five epochs when TRish and SG are employed to optimize the convolutional neural network using the `cifar10` dataset.

5 Conclusion

An algorithm inspired by a trust region methodology has been proposed, analyzed, and tested for solving stochastic and finite-sum minimization problems. Our proved theoretical guarantees show that our method, deemed TRish, has convergence properties that are similar to a traditional SG method. Our numerical results, on the other hand, show that TRish can outperform a traditional SG approach. We attribute this better behavior to the algorithm's use of normalized steps, which one can argue lessens its dependence on problem-specific quantities.

Naturally, a more substantial numerical study—that goes well beyond the scope of this paper—would be necessary to fully explore the trade-offs between TRish and SG in practice. For example, a more substantial numerical study would take into account different procedures that might be used to decrease the stepsize after some number of iterations, as is typically done in practice. Indeed, for the convex problems that we considered, this was our motivation for presenting results for only one epoch, since, in practice, one often adjusts the stepsize after each epoch. For TRish, this adjustment may involve updates to the pair (γ_1, γ_2) as well, which one might adjust so that $\gamma_1 - \gamma_2 = \mathcal{O}(\alpha)$, as our theory suggests.

Finally, while not considered in this paper, we believe it would be interesting to explore the incorporation within TRish of various enhancements, such as the use of second-derivative (i.e., Hessian) approximations, acceleration, and/or momentum. These might further improve the practical performance of the framework set forth in this paper.

Acknowledgment

The authors would like to thank Chaoxu Zhou of Columbia University for his valuable assistance in correcting errors in an earlier version of this manuscript. They would also like to thank the anonymous referees and the Associate Editor for providing valuable comments on an earlier draft of the paper. In particular, one anonymous referee suggested a proof for Lemma 3.2, which greatly improved the analysis. All authors were supported by the U.S. National Science Foundation's Division of Computing and Communication Foundations and Division of Mathematical Sciences under award numbers CCF–1618717 and DMS–1319356. The first author was also supported by the U.S. Department of Energy, Office of Science, Applied Mathematics, Early Career Research Program under Award Number DE–SC0010615.

References

[1] A. Agarwal and L. Bottou. A lower bound for the optimization of finite sums. In *Proceedings of the International Conference on Machine Learning*, volume 37, pages 78–86. PMLR, 2015.

[2] L. Bottou, F. E. Curtis, and J. Nocedal. Optimization Methods for Large-Scale Machine Learning. *SIAM Review*, 60(2):223–311, 2018.

[3] R. H. Byrd, G. M. Chin, J. Nocedal, and Y. Wu. Sample Size Selection in Optimization Methods for Machine Learning. *Mathematical Programming, Series B*, 134(1):127–155, 2012.

[4] Chih-Chung Chang and Chih-Jen Lin. LIBSVM: A library for support vector machines. *ACM Transactions on Intelligent Systems and Technology*, 2:27:1–27:27, 2011. Software available at http://www.csie.ntu.edu.tw/~cjlin/libsvm.

[5] Ruobing Chen, Matt Menickelly, and Katya Scheinberg. Stochastic optimization using a trust-region method and random models. *Mathematical Programming*, 169(2):447–487, 2018.

[6] K. L. Chung. On a stochastic approximation method. *Annals of Mathematical Statistics*, 25(3):463–483, 1954.

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

[8] J. E. Freund. *Mathematical Statistics.* Prentice-Hall, Englewood Cliffs, NJ, USA, 1962.

[9] M. P. Friedlander and M. Schmidt. Hybrid deterministic-stochastic methods for data-fitting. *SIAM Journal on Scientific Computing*, 34:A1380–A1405, 2012.

[10] Saeed Ghadimi and Guanghui Lan. Stochastic first- and zeroth-order methods for nonconvex stochastic programming. *SIAM Journal on Optimization*, 23(4):2341–2368, 2013.

[11] E. G. Gladyshev. On stochastic approximations. *Theory of Probability and its Applications*, 10:275–278, 1965.

[12] Elad Hazan, Kfir Levy, and Shai Shalev-Shwartz. Beyond convexity: Stochastic quasi-convex optimization. In *Advances in Neural Information Processing Systems*, pages 1594–1602, 2015.

[13] Rie Johnson and Tong Zhang. Accelerating stochastic gradient descent using predictive variance reduction. In *Advances in Neural Information Processing Systems*, pages 315–323, 2013.

[14] Hamed Karimi, Julie Nutini, and Mark Schmidt. Linear convergence of gradient and proximal-gradient methods under the polyak-łojasiewicz condition. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pages 795–811. Springer, 2016.

[15] A. Krizhevsky. Learning multiple layers of features from tiny images. Technical Report, University of Toronto, 2009.

[16] Jeffrey Larson and Stephen C Billups. Stochastic derivative-free optimization using a trust region framework. *Computational Optimization and Applications*, 64(3):619–645, 2016.

[17] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner. Gradient-based learning applied to document recognition. In *Proceedings of the IEEE, 86(11)*, pages 2278–2324, 1998.

[18] A. Nemirovski, A. Juditsky, G. Lan, and A. Shapiro. Robust Stochastic Approximation Approach to Stochastic Programming. *SIAM Journal on Optimization*, 19(4):1574–1609, 2009.

[19] H. Robbins and S. Monro. A Stochastic Approximation Method. *The Annals of Mathematical Statistics*, 22(3):400–407, 1951.

[20] H. Robbins and D. Siegmund. A convergence theorem for nonnegative almost supermartingales and some applications. In Jagdish S. Rustagi, editor, *Optimizing Methods in Statistics*. Academic Press, 1971.

[21] Stéphane Ross, Paul Mineiro, and John Langford. Normalized online learning. In *Proceedings of the Twenty-Ninth Conference on Uncertainty in Artificial Intelligence (UAI)*, 2013.

[22] A. Shapiro, D. Dentcheva, and A. Ruszczyński. *Lectures on Stochastic Programming: Modeling and Theory*. SIAM, 2009.

[23] Yung Liang Tong. *The multivariate normal distribution*. Springer Science & Business Media, 2012.