Journal of Computational Mathematics
Vol.43, No.5, 2025, 1169–1193.

http://www.global-sci.org/jcm
doi:10.4208/jcm.2504-m2023-0228

AN ACCELERATED STOCHASTIC TRUST REGION METHOD FOR STOCHASTIC OPTIMIZATION*

Rulei Qi, Dan Xue[1], Jing Li and Yujia Zhai
School of Mathematics and Statistics, Qingdao University, Qingdao 266071, China
Emails: 15865589629@163.com, wtxuedan@126.com,
2021020253@qdu.edu.cn, 838090634@qq.com

Abstract

In this paper, we propose an accelerated stochastic variance reduction gradient method with a trust-region-like framework, referred as the NMSVRG-TR method. Based on NMSVRG, we incorporate a Katyusha-like acceleration step into the stochastic trust region scheme, which improves the convergence rate of the SVRG methods. Under appropriate assumptions, the linear convergence of the algorithm is provided for strongly convex objective functions. Numerical experiment results show that our algorithm is generally superior to some existing stochastic gradient methods.

Mathematics subject classification: 65K05, 90C15.
Key words: Stochastic optimization, Stochastic variance reduced gradient, Trust region, Gradient descent method, Machine learning.

1. Introduction

In this paper, we consider stochastic optimization problems, in which random variables are used to describe uncertainties in the objective function. The general stochastic optimization model can be summarized as follows:

$$\min_{x\in\mathbb{R}^n} f(x)=\mathbb{E}_\xi[f(x;\xi)], \tag{1.1}$$

where ξ is a random variable and $\mathbb{E}_\xi[]$ represents the mathematical expectation of ξ. In many practical applications, due to the real distribution of ξ is unknown, we generally use ξ empirical distribution instead of the actual distribution. Specifically, we assume that there are n samples $\xi_1,\xi_2,\ldots,\xi_n$, let $f_i(x)=f(x;\xi_i)$, and then obtain a finite-sum structure

$$\min_{x\in\mathbb{R}^n} f(x)=\frac{1}{n}\sum_{i=1}^{n} f_i(x), \tag{1.2}$$

where x is the decision variable, n is the sample size, and each $f_i:\mathbb{R}^n\to\mathbb{R}$ is a loss function that corresponds to the i-th data sample. In machine learning and statistics, many regularized empirical risk minimization (ERM) problems are commonly expressed as a finite-sum structure [3, 5, 13]. We assume f_i is convex with a Lipschitz continuous gradient, and the function $f(x)$ is smooth and strongly convex.

* Received October 14, 2023 / Revised version received November 1, 2024 / Accepted April 3, 2025 / Published online May 16, 2025 /

[1] Corresponding author

In many applications, it is challenging to compute the full gradient $\nabla f(x)$ when the collected sample size n is enormous. Therefore, in recent years, how to solve the problems (1.2) more efficiently has attracted widespread attention from many scholars.

Stochastic gradient descent (SGD) can be traced back to the pioneering work of [19]. Utilizing the separable structure of $f(x)$, SGD estimates the full gradient $\nabla f(x)$ of the current iteration by utilizing only one component of the gradient $\nabla f_{i_t}(x), i_t$ is a uniformly randomly selected metric from $\{1, 2, \ldots, n\}$. The iteration format is as follows:

$$x_{t+1} = x_t - \alpha_t \nabla f_{i_t}(x_t), \tag{1.3}$$

where $\alpha_t > 0$ denotes the step size and $\nabla f_{i_t}(x_t)$ is defined as the stochastic gradient at x_t. In most cases, the cost of each iteration of the SGD method is $1/n$ of the full gradient descent method [9], which shows that the complexity of each iteration of the stochastic gradient descent method is significantly lower than that of full gradient descent. Therefore, SGD scales well in data sample size, which is important in some machine learning applications because there are many large data samples.

The more common form in practical calculations is the mini-batch stochastic gradient method. It has the following iteration format:

$$x_{t+1} = x_t - \frac{\alpha_t}{|\mathcal{I}_t|} \sum_{i \in \mathcal{I}_t} \nabla f_i(x_t), \tag{1.4}$$

where $\mathcal{I}_t$ is an index chosen uniformly at random from $\mathcal{I}_t \in \{1, 2, \ldots, n\}$, and $|\mathcal{I}_t|$ represents the number of elements in the $\mathcal{I}_t$.

Nonetheless, it is essential to note that the variance estimate linked to the stochastic gradient, produced by the stochastic gradient descent, persists throughout the iterations. This shows that SGD needs to reduce the step size to achieve convergence, as discussed in reference [4]. Even in the convex case, this algorithm can only show the sublinear convergence rate, as described in reference [24]. As a result, the optimization community has shown a keen interest in strategies to decrease the variance and accelerate the convergence rate.

The stochastic average gradient (SAG) method, proposed by Roux *et al.* [14], records all previously obtained stochastic gradients. These gradients are subsequently averaged with the concurrently updated stochastic gradient, serving as the gradient estimation for the next iteration. The iteration format is as follows:

$$x_{t+1} = x_t - \alpha_t \left(\frac{1}{n} (\nabla f_{i_t}(x_t) - g_{i_t}^{t-1}) + \frac{1}{n} \sum_{i=1}^{n} g_i^{t-1} \right), \tag{1.5}$$

where g_i^t is updated by

$$g_i^t = \begin{cases} \nabla f_{i_t}(x_t), & \text{if} \quad i = i_t, \\ g_i^{t-1}, & \text{otherwise.} \end{cases} \tag{1.6}$$

SAG is linearly convergent with respect to smooth strongly convex functions. It is crucial to acknowledge that the SAG gradient estimation is biased. Subsequently, Defazio [8] modified SAG and proposed SAGA algorithm. This method employs an unbiased gradient estimation for the update direction and converges at the same rate as SAG. The SAGA gradient estimation method can be understood as retaining a history of n past stochastic gradients and updating these stored gradients

$$x_{t+1} = x_t - \alpha_t \left(\nabla f_{i_t}(x_t) - g_{i_t}^{t-1} + \frac{1}{n} \sum_{i=1}^{n} g_i^{t-1} \right). \tag{1.7}$$