CARPATHIAN J. MATH.
Volume **41** (2025), No. 3,
Pages 849-862

Online version at https://www.carpathian.cunbm.utcluj.ro/
Print Edition: ISSN 1584 - 2851; Online Edition: ISSN 1843 - 4401
DOI: https://doi.org/10.37193/CJM.2025.03.14

Efficient Reduction of Variances in Stochastic Spectral Conjugate Gradient Algorithm

SEIFU ENDRIS YIMER[1], POOM KUMAM[2,*], AND PARIN CHAIPUNYA[3]

ABSTRACT. Conjugate gradient methods are often popular for solving nonlinear optimization problems. In this paper, we discuss the spectral conjugate gradient (SCG) method, an effective numerical method that generalizes the conjugate gradient method (CG) for solving a large-scale unconstrained optimization problem. Integrating the methods of Fletcher and Reeves (FR), and Polak and Ribiere (PR), we introduce a new stochastic spectral conjugate gradient algorithm with variance reduction, and we show that it is linearly convergent with the Fletcher and Reeves method for smooth and strongly convex functions. Thus, we illustrate experimentally that our algorithm converges quicker than its companions for the four learning models considered. Moreover, likewise the CG method, it only stores the last gradient vector so that it would be easy to apply and handle some complex problems considered as in machine learning. In the experiment, we also show that our algorithm overtakes generalization performance (AUC) over their corresponding companions through the four models considered that might be nonsmooth or nonconvex.

1. INTRODUCTION

Consider the following finite sum optimization problem which has been largely discussed for solving most machine learning problems;

$$\min_{w} \phi(w) = \sum_{i=1}^{n} \phi_i(w) \tag{1.1}$$

where $w \in \mathbb{R}^d$, and $\phi_i : \mathbb{R}^d \to \mathbb{R}, i = 1, ..., m$ which indeed plays an indispensable role in estimating the best parameter w that fits the training data in their respective i-th sample. It can be considered as the most extensively used approach to solve some of the well known machine learning problems; classification[14], Regression[35], Clustering[17], and Ranking[16, 15], One of the most popular and ancient algorithms used for minimizing (1.1) in many large-scale machine learning problems is known as the stochastic gradient descent (SGD)[4] and its variants [7, 27]. A lot has been done on this algorithmic approach such as the Adam method[20], that evaluates the adaptive learning rate for their corresponding parameters, SGD with momentum[31] is the other well known method which can be used to train both recurrent and deep neural networks by considering a random initialization and a relatively slow increase of the momentum parameter. However, the method SGD basically relies on the initialization and decay strategy of the learning rate.

Due to randomness, SGD often introduces a variance that drops down its rate of convergence. To alleviate this problem, Roux et al.[28] introduced the stochastic average gradient for effective reduction of the variances in SGD. The stochastic dual coordinate ascent (SDCA) was introduced by Shalev-Shwartz and Zhang [30] for the training of some linear

Received: 24.07.2024. In revised form: 07.01.2025. Accepted: 16.03.2025
2020 *Mathematics Subject Classification.* 49M07, 49M10, 90C06, 65K05.
Key words and phrases. *spectral conjugate gradient, large-scale unconstrained optimization, stochastic quasi-Newton method, non-convex stochastic optimization, Unconstrained optimization, Conjugate gradient method.*
Corresponding author: *Poom Kumam; e-mail: poom.kum@kmutt.ac.th

prediction problems with a favorable linear rate of convergence. However, these methods require the storage of all gradients which make them impractical for a large complex problem. By reducing the variance of the gradient estimate Johnson and Zhang [19] proposed a stochastic variance reduced gradient (SVRG) to accelerate the convergence of the corresponding first-order stochastic method. Moreover, some works have also been indicated as a promising line of research by stochasticizing the second order quasi-Newton method through L-BFGS algorithm. Besides, stochastic quasi-Newton method for the non-convex stochastic optimization was studied by Xiao Wang *et al*. [33]. To determine the descent direction and approximate the curvature of the objective function, Mokhtari and Ribeiro made use of stochastic gradients instead of the deterministic gradients [24]. The stochastic variants of L-BFGS was introduced by Moritz *et al* [25]who integrated the idea of variance reduction. In a limited memory, Gower *et al* [11] introduced a stochastic block BFGS update with SVRG approach. Nevertheless, computing the product $H\nabla f$ (H is the Hessian) within a limited memory may become restrictive for large scale machine learning problems due to the fact that the stochastic quasi-Newton methods often require m vector pairs to evaluate it.

The FR method[9], an extension of a linearly CG method to their corresponding nonlinear functions, was first discussed by Fletcher and Reeves. Another variant of the CG method called PR method was introduced by Polak and Ribiere (PR)[26] and was further modified by Gilbert and Nocedal showing that under a sufficient descent condition, the modified PR method $\beta_t^{PR+} = \max\{\beta_t^{PR}, 0\}$ with Wolfe-Powell linear search is globally convergent. In a situation where the direction is violated, restarting is typically essential upon which the PR method, Hestenes and Stiefel [12] method, and Liu and Storey[23] methods became the most efficient and suitable CG methods for practical implementation. Even though the convergence of the conjugate descent [6], Dai and Yuan [5] method, and the FR method which are indeed poor in their numerical results. The spectral conjugate gradient method (SCG) is remarkably and essentially considered as a generalization of the CG method for solving large-scale unconstrained optimization problems. It was first proposed by Birgin and Martinez([3]) and a lot of improvement with excellent theoretical and numerical results have been proposed, see references there in [2, 22, 13, 34, 10, 32, 1]. Recently, Xiao-Bo *et al* [18] and Caixia Kou *et al* [21] proposed variants of stochastic CG methods which overtake some advantages one over the other in terms of calculating and handling the variance of the stochastic gradient in charge. Particularly, inspired by the work of Xiao-Bo *et al* called the stochastic conjugate gradient algorithm with variance reduction(CGVR), in this paper therefore, we present a novel scheme that generalizes their works by introducing the appropriate spectral parameter that best suits for searching the direction of descent at each iteration.The proposed algorithms, namely, the stochastic Spectral Fletcher-Reeves Conjugate Gradient Variance Reduction (SFRCGVR) and the stochastic Spectral Polyak and Ribiere Conjugate Gradient Variance Reduction (SPRCGVR) have the following comparative advantages:

- Both converge rapidly within a few iterations due to the concept of SVRG and CGVR.
- The presence of the spectral parameter would make our algorithms converge better than their counterparts because the spectral CG converges faster than the general CG.
- The FR variance reduced spectral CG method (SFRCGVR) as well as PR variance reduced spectral CG method (SPRCGVR) work well because the parameters are insensitive to the data.

- Unlike the quasi-Newton variants that often store a set of vector pairs, it only stores the last gradient vector similar to CG

Hence, the contributions of our paper are to propose an extended stochastic variant of the CG method with variance reduction on the subsamples where computation of the gradient and the Wolfe line search are consequently evaluated. Furthermore, we show the linear convergence of SFRCGVR for strongly convex and smooth functions with the FR method. We also demonstrate a series of experiments on five large scale data sets with four classic learning models which can be convex, nonconvex, or nonsmooth. The experiment shows that our algorithms converge quicker than their counterparts. In particular its AUC (area under the curve) performance with the sqhinge loss model is comparable to that of the LIBLINEAR solver with a remarkable improvement in computational efficiency.

In the composition of this paper, we discuss the proposed algorithms and their frameworks briefly in section II and III respectively. We show the convergence analysis of our algorithms (SFRCGVR/SPRCGVR) in the next section and show its linear convergence for a strongly convex and smooth function. Finally, we present and compare the experimental results of our algorithms along with their counterparts in section IV, and conclude some results and related figures in the last section as well.

2. Stochastic Spectral Conjugate Gradient with Variance Reduction

Even though SVRG uses a gradient estimate using the variance reduction approach to accelerate the convergence of SGD, it is very sensitive to the learning rate. However, SLBFGS often requires M vector pairs to determine $H\nabla f$ and hence it is often computationally expensive. Stochastic conjugate gradient with Variance Reduction algorithm (CGVR) was proposed to alleviate the above problems (see Algorithm 1). There are two loops that need to be considered in Algorithm 1, the outer and the inner loops. The data points are chosen randomly in the inner loop to estimate a gradient with the variance reduction strategy. Motivated by CGVR, we introduce a new scheme for the choice of the spectral and conjugate parameters to improve the performance and computational efficiency of the algorithm.

3. Main result

The stochastic variance reduced gradient algorithm (SVRG) shows better convergence property than SGD due to the reduction of variance in the gradient estimate; however, it is sensitive to the learning rate as well. We incorporate the spectral CG algorithm and SVRG to get our algorithm. Two loops including the outer and inner loops are considered in Algorithm 2. Likewise CGVR, we compute the full gradient v_k in the outer loop and retain x_0 to initiate w after every m SGD iteration. The variance reduced gradient g_{t+1} is computed on a randomly generated set $Q_{k,t} \subset \{1, 2, ..., n\}$ in the t-th loop of the k-th iteration, i.e.,

$$g_{t+1} = \nabla\phi_{Q_{k,t}}(x_{t+1}) - \nabla\phi_{Q_{k,t}}(x_0) + v_k, \tag{3.2}$$

where g_{t+1} corresponds to $\nabla\phi(x_{t+1})$ in CG, and $\nabla\phi_{Q_{k,t}}(.)$ is calculated from the definition of the subsampled function

$$\phi_Q(w) = \frac{1}{|Q|}\sum_{w\in Q}\phi_i(w). \tag{3.3}$$

Algorithm 1 : Stochastic Spectral Conjugate Gradient with Variance Reduction

Initialize: w_0 and $l_0 = \nabla\phi(w_0)$
for $k = 1, 2, ...P-1$ do
  $v_k = \nabla\phi(w_k)$
  $x_0 = w_k$
  $g_0 = l_k$
  $d_0 = -g_0$
  for $t = 0, ..., m-1$ **do**
    randomly pick a minibatch $Q_{k,t} \subset \{1, 2, ..., n\}$
    find α_t using the line search algorithm to optimize:
      $\min_\alpha \phi_{Q_{k,t}}(x_t + \alpha d_t)$
    $x_{t+1} = x_t + \alpha_t d_t$
    Compute $g_{t+1} = \nabla\phi_{Q_{k,t}}(x_{t+1}) - \nabla\phi_{Q_{k,t}}(x_0) + v_k$
    find β and θ in the following two options
    Option I:
      $\beta_{t+1}^{PR} = \frac{g_{t+1}^T(g_{t+1}-g_t)}{\|g_t\|^2}, \quad \theta_{t+1} = \frac{d_t^T y_t}{\|g_t\|^2}$
    Option II:
      $\beta_{t+1}^{FR} = \frac{\|g_{t+1}\|^2}{\|g_t\|^2}, \quad \theta_{t+1} = \frac{d_t^T y_t}{\|g_t\|^2},$
    $d_{t+1} = -\theta_{t+1} g_{t+1} + \beta_{t+1} d_t$
  end for
  $l_{k+1} = g_m$
  Option I: $w_{k+1} = x_m$
  Option II: $w_{k+1} = x_t$ for randomly chosen $t \in \{0, 1, 2, ..., m-1\}$
end for

A line search algorithm is called to optimize its step size so that the iterations in the inner loop can be easily computed using a precise formulations of the parameters in the classical spectral conjugate gradient algorithms, as can be seen in Algorithm 2. Two options are considered to choose w_{k+1} (see Option I and Option II in Algorithm 2).
The PR method is a significant CG method with a parameter β_{t+1} defined as

$$\beta_{t+1}^{PR} = \frac{g_{t+1}^T(g_{t+1} - g_t)}{\|g_t\|^2}. \tag{3.4}$$

FR methods use another approach to compute β_{t+1} as follows

$$\beta_{t+1}^{FR} = \frac{\|g_{t+1}\|^2}{\|g_t\|^2}. \tag{3.5}$$

Particularly, these parameters together with the spectral parameter

$$\theta_{t+1} = \frac{d_t^T y_t}{\|g_t\|^2}, \quad \text{where} \quad y_t = g_{t+1} - g_t \tag{3.6}$$

are used to compute the direction(descent) towards the solution in context see [8] under a certain conditions. In our experiment, we used the trick $\beta_{t+1} = 0$ to restart the iteration with the steepest descent step and computed the parameter β as in the PR^+ method,

$$\beta_{t+1}^{PR+} = \max\left\{\beta_{t+1}^{PR}, 0\right\}. \tag{3.7}$$

and restarting periodically will enhance the algorithm to refresh. Notice that the step size α_t satisfying the following Wolf type conditions[8] will ensure that the search direction d_t is descent.

$$\phi(x_t) - \phi(x_t + \alpha_t d_t) \geq \rho\alpha_t^2 \|d_t\|^2, \tag{3.8}$$

$$g(x_t + \alpha_t d_t)^T d_t \geq -2\sigma\alpha_t \|d_t\|^2, \tag{3.9}$$

where $0 < \rho < \sigma < 1$.
In fact, the ideal optimal step length can be obtained using exact line search method;

$$\min_{\alpha} \phi_{Q_{k,t}}(x_t + \alpha d_t). \tag{3.10}$$

4. Convergence Analysis

Let us define

$$\pi_\phi(x) = \phi(x) - \phi(w_*), \phi_t = \phi(x_t), \nabla\phi_t = \nabla\phi(x_t)$$

Let's see the convergence of our algorithm with a Fletcher-Reeves parameter β_t update (3.5)(Option II). In this paper, we frequently use β_t to denote β_t^{FR} unless stated, otherwise. Based on the assumptions in[8], we have the following Wolfe type line search to ensure the convergence of nonlinear CG methods with inexact line search

$$\phi(x_t) - \phi(x_t + \alpha_t d_t) \geq \rho\alpha_t^2 \|d_t\|^2 \tag{4.11}$$

$$g(x_t + \alpha_t d_t)^T d_t \geq -2\sigma\alpha_t \|d_t\|^2 \tag{4.12}$$

which in turn implies

$$(2\sigma + L)\alpha_t \|d_t\|^2 \geq -g_t^T d_t, \tag{4.13}$$

where $0 < \rho < \sigma < 1$ and L is a Lipschitz constant.
Our analysis uses the following assumptions.

Assumption 1
For a twice continuously differentiable function ϕ_i, there are positive constants $\gamma < \Gamma$ such that

$$\gamma I \leq \nabla^2 \phi_Q(x) \leq \Gamma I, \ \forall x \in \mathbb{R}^d \tag{4.14}$$

where $Q \subset \{1, 2, ..., n\}$
Assumption 2
There is a constant $\mu < 1$, such that

$$\beta_t^{FR} = \frac{\|g_t\|^2}{\|g_{t-1}\|^2} \leq \mu. \tag{4.15}$$

Assumption 3
Algorithm 2 is consistent with a step length $\alpha_t \in [a, b]$ such that $b > a > 0$.

As can be seen in Lemma 5 and 6 of [25], we need to estimate the lower and upper bounds of $\|\nabla\phi(x)\|$ and $\mathbb{E}[\|g_t\|^2]$ respectively. Note that

$$\nabla\phi(x_t) = \mathbb{E}[g_t]. \tag{4.16}$$

Lemma 4.1. *Let the function ϕ be $\gamma-$ strongly convex and continuously differentiable. Suppose a minimizer w_* is unique, then, we get*

$$\|\nabla\phi(x)\|^2 \geq 2\gamma\big(\phi(x) - \phi(w_*)\big), \quad \forall x \in \mathbb{R}^d \tag{4.17}$$

Lemma 4.2. *Let $g_t = \nabla\phi_{Q_{k,t}}(x_t) - \nabla\phi_{Q_{k,t}}(w_k) + v_k$ and $v_k = \nabla\phi(w_k)$ be the stochastic variance reduced gradient. Suppose w_* be the unique minimizer of ϕ. Then, we have*

$$\mathbb{E}[\|g_t\|^2] \leq 4\Gamma\big(\pi_\phi(x_t) + \pi_\phi(w_k)\big), \tag{4.18}$$

where

$$\pi_\phi(x) = \phi(x) - \phi(w_*).$$

Lemma 4.3. *Suppose the direction d_t is given* [8]*, then we have*

$$g_t^T d_t = -\|g_t\|^2. \tag{4.19}$$

Thus, we obtain the main results as follows.

Theorem 4.1. *Let assumptions* 2 *and* 3 *hold to algorithm 1. Then , we have*

$$\mathbb{E}[\|d_t\|^2] \leq \varphi(t)\mathbb{E}[\|g_0\|^2], \tag{4.20}$$

where

$$\varphi(t) = \frac{M}{1-\mu}\mu^t - \frac{M+\mu-1}{1-\mu}\mu^{2t} \quad \textit{and} \quad M = \frac{2\sigma+L}{2a\rho\gamma} \tag{4.21}$$

Proof. According to Assumption 2, we have

$$\mathbb{E}[\|g_t\|^2] \leq \mu\mathbb{E}[\|g_{t-1}\|^2]. \tag{4.22}$$

and a bounded $\mathbb{E}[\|d_t\|^2]$,

$$\begin{aligned}
\mathbb{E}[\|d_t\|^2] &= \mathbb{E}[\| - \theta_t g_t + \beta_t d_{t-1}\|^2] \\
&= \mathbb{E}[\theta_t^2\|g_t\|^2 - 2\theta_t\beta_t g_t^T d_{t-1} + \beta_t^2\|d_{t-1}\|^2] \\
&= \mathbb{E}[\theta_t^2\|g_t\|^2 - 2\theta_t\|g_t\|^2\big(\theta_t - 1\big) + \beta_t^2\|d_{t-1}\|^2] \\
&= \mathbb{E}[\beta_t^2\|d_{t-1}\|^2 + 2\theta_t\|g_t\|^2 - \theta_t^2\|g_t\|^2] \\
&= \mathbb{E}[\beta_t^2\|d_{t-1}\|^2 - \|g_t\|^2(\theta_t^2 - 2\theta_t)] \\
&= \mathbb{E}[\beta_t^2\|d_{t-1}\|^2 - \|g_t\|^2(\theta_t - 1)^2 + \|g_t\|^2] \\
&\leq \mathbb{E}[\beta_t^2\|d_{t-1}\|^2 + \|g_t\|^2] \\
&= \mathbb{E}[\beta_t^2\|d_{t-1}\|^2 - g_t^T d_t] \\
&\leq \mathbb{E}[\beta_t^2\|d_{t-1}\|^2 + (2\sigma+L)\alpha_t\|d_t\|^2] \\
&= \mathbb{E}[\beta_t^2\|d_{t-1}\|^2 + \frac{(2\sigma+L)}{\rho\alpha_t}\rho\alpha_t^2\|d_t\|^2] \\
&\leq \mathbb{E}[\beta_t^2\|d_{t-1}\|^2 + \frac{(2\sigma+L)}{\rho\alpha_t}\big(\phi(x_t) - \\
&\quad \phi(x_t + \alpha_t d_t)\big)] \\
&\leq \mathbb{E}[\beta_t^2\|d_{t-1}\|^2 + \frac{(2\sigma+L)}{\rho\alpha_t}\frac{1}{2\gamma}\|g_t\|^2] \\
&= \mathbb{E}[\beta_t^2\|d_{t-1}\|^2 + M\|g_t\|^2] \\
&= \beta_t^2\mathbb{E}[\|d_{t-1}\|^2] + M\mathbb{E}[\|g_t\|^2]. \qquad (4.23)
\end{aligned}$$

Thus,

$$\mathbb{E}[\|d_t\|^2] \leq \mu^2\mathbb{E}[\|d_{t-1}\|^2] + M\mu\mathbb{E}[\|g_{t-1}\|^2]. \tag{4.24}$$

where

$$M = \frac{2\sigma+L}{2a\rho\gamma}.$$

To begin with the k-th iteration, we have

$$d_0 = -g_0. \tag{4.25}$$

Furthermore, we unfold (4.22) inductively as

$$\mathbb{E}[\|g_t\|^2] \leq \mu^t \mathbb{E}[\|g_0\|^2]. \tag{4.26}$$

According to (4.24) and (4.26), we get the following

$$\begin{aligned}
\mathbb{E}[\|d_t\|^2] &\leq M\mu\mathbb{E}[\|g_{t-1}\|^2] + \mu^2\mathbb{E}[\|d_{t-1}\|^2] \\
&\leq M\mu\big(\mathbb{E}[\|g_{t-1}\|^2] + \mu^2\mathbb{E}[\|g_{t-2}\|^2] \\
&\quad + \ldots + (\mu^2)^{t-1}\mathbb{E}[\|g_0\|^2]\big) + (\mu^2)^t\mathbb{E}[\|d_0\|^2] \\
&= M\mu\big(\mu^{t-1}\mathbb{E}[\|g_0\|^2] + (\mu^2)\mu^{t-2}\mathbb{E}[\|g_0\|^2] \\
&\quad + \ldots + (\mu^2)^{t-1}\mathbb{E}[\|g_0\|^2]\big) + (\mu^2)^t\mathbb{E}[\|g_0\|^2] \\
&= M\mu\mathbb{E}[\|g_0\|^2]\sum_{j=0}^{t-1}(\mu^2)^j\mu^{t-1-j} + (\mu^2)^t\mathbb{E}[\|g_0\|^2] \\
&= M\mu^t\mathbb{E}[\|g_0\|^2]\sum_{j=0}^{t-1}\mu^j + (\mu^2)^t\mathbb{E}[\|g_0\|^2] \\
&= \left(M\mu^t\frac{1-\mu^t}{1-\mu} + \mu^{2t}\right)\mathbb{E}[\|g_0\|^2] \\
&= \left(\frac{M}{1-\mu}\mu^t - \frac{M+\mu-1}{1-\mu}\mu^{2t}\right)\mathbb{E}[\|g_0\|^2] \\
&= \varphi(t)\mathbb{E}[\|g_0\|^2], \qquad (4.27)
\end{aligned}$$

where

$$\varphi(t) = \frac{M}{1-\mu}\mu^t - \frac{M+\mu-1}{1-\mu}\mu^{2t} \quad \text{and} \quad M = \frac{2\sigma+L}{2a\rho\gamma}. \tag{4.28}$$

□

Theorem 4.2. *Suppose that w_* be the unique minimizer of ϕ under the assumptions 1,2 and 3 , then we have*

$$\mathbb{E}[\phi(w_k) - \phi(w_*)] \leq \xi^k\mathbb{E}[\phi(w_0) - \phi(w_*)], \forall k \geq 0, \tag{4.29}$$

where

$$\xi = \frac{4\Gamma^2b^2(M+1) + (1-\mu)^2}{2a\gamma m(1-\mu)^2} < 1 \tag{4.30}$$

is its rate of convergence, and thus for a very large m to hold

$$m \geq \frac{4\Gamma^2b^2(M+1) + (1-\mu)^2}{2a\gamma(1-\mu)^2} > 0. \tag{4.31}$$

Proof. From Lipscitz continuity of $\nabla\phi$ and Assumption 2, we get

$$\phi(x_{t+1}) \leq \phi(x_t) + \nabla\phi(x_t)^T(x_{t+1} - x_t) + \frac{\Gamma}{2}\|x_{t+1} - x_t\|^2. \tag{4.32}$$

Note that $d_0 = -g_0$, then we have $\nabla\phi_0^T\mathbb{E}[d_0] = -\|\nabla\phi_0\|^2$. Since $d_t = -\theta_t g_t + \beta_t d_{t-1}$ for $t \geq 0$ and the random variables g_t and d_{t-1} are independent, with (4.16) and (4.19), we

have

$$\begin{aligned}\nabla\phi^T\mathbb{E}[d_t] &= \mathbb{E}[g_t]\mathbb{E}[d_t]\\ &= \mathbb{E}[g_t^T d_t]\\ &= -\mathbb{E}\|g_t\|^2\\ &= -\|\nabla\phi_t\|^2. \end{aligned} \tag{4.33}$$

So, we can take expectation on both sides of (4.32)

$$\begin{aligned}\mathbb{E}[\phi_{t+1}] &\leq \phi_t + \nabla\phi_t^T\mathbb{E}[(x_{t+1}-x_t)] + \frac{\Gamma}{2}\mathbb{E}[\|x_{t+1}-x_t\|^2]\\ &= \phi_t + \alpha_t\nabla\phi_t^T\mathbb{E}[d_t] + \frac{\Gamma}{2}\alpha_t^2\mathbb{E}[\|d_t\|^2]\\ &= \phi_t - \alpha_t\|\nabla\phi_t\|^2 + \frac{\Gamma}{2}\alpha_t^2\mathbb{E}[\|d_t\|^2]\\ &\leq \phi_t - a\|\nabla\phi_t\|^2 + \frac{\Gamma}{2}b^2\varphi(t)\mathbb{E}[\|d_0\|^2]\\ &\leq \phi_t - a\|\nabla\phi_t\|^2 + \frac{\Gamma}{2}b^2\varphi(t).4\Gamma(\pi_\phi(x_0)+\pi_\phi(w_k))\\ &\leq \phi_t - 2a\gamma\pi_\phi(x_t) + \tau\varphi(t)\pi_\phi(w_k), \end{aligned} \tag{4.34}$$

where

$$\tau = 4\Gamma^2b^2 \quad \text{and} \quad \varphi(t) = \frac{M}{1-\mu}\mu^t - \frac{M+\mu-1}{1-\mu}\mu^{2t}.$$

A telescoping sum over $t = 0, 1, 2, ..., m-1$ gives

$$\begin{aligned}\mathbb{E}[\phi_m] \quad &\leq \quad \mathbb{E}[\phi_0] - 2a\gamma\left(\sum_{t=0}^{m-1}\mathbb{E}[\pi_\phi(x_t)]\right)\\ &\quad + \tau\mathbb{E}[\pi_\phi(w_k)]\sum_{t=0}^{m-1}\varphi(t). \end{aligned} \tag{4.35}$$

Computing $\sum_{t=0}^{m-1}\varphi(t)$, we get

$$\begin{aligned}\sum_{t=0}^{m-1}\varphi(t) &= \sum_{t=0}^{m-1}\left(\frac{M}{1-\mu}\mu^t - \frac{M+\mu-1}{1-\mu}\mu^{2t}\right)\\ &= \frac{M}{1-\mu}\left(\frac{1-\mu^m}{1-\mu}\right) - \frac{M+\mu-1}{1-\mu}\left(\frac{1-\mu^{2m}}{1-\mu^2}\right)\\ &\leq \left(\frac{M\mu}{(1+\mu)(1-\mu)^2} + \frac{1}{1-\mu^2}\right)(1-\mu^{2m})\\ &\leq \left(\frac{M\mu}{1+\mu}+1\right)\frac{1-\mu^{2m}}{(1-\mu)^2}\\ &\leq \frac{M+1}{(1-\mu)^2}. \end{aligned} \tag{4.36}$$

Then from (4.35), we have

$$
\begin{aligned}
0 &\leq \mathbb{E}[\phi_0] - \mathbb{E}[\phi_m] - 2a\gamma m\mathbb{E}[\pi_\phi(w_{k+1})] \\
&\quad + \tau\frac{M+1}{(1-\mu)^2}\mathbb{E}[\pi_\phi(w_k)] \qquad (4.37)\\
&\leq \mathbb{E}[\phi(w_k) - \phi(w_*)] - 2a\gamma m\mathbb{E}[\pi_\phi(w_{k+1})] \\
&\quad + \tau\frac{M+1}{(1-\mu)^2}\mathbb{E}[\pi_\phi(w_k)] \qquad (4.38)\\
&= \mathbb{E}[\pi_\phi(w_k)] - 2a\gamma m\mathbb{E}[\pi_\phi(w_{k+1})] \\
&\quad + \tau\frac{M+1}{(1-\mu)^2}\mathbb{E}[\pi_\phi(w_k)] \qquad (4.39)\\
&= \left(1 + \frac{\tau(M+1)}{(1-\mu)^2}\right)\mathbb{E}[\pi_\phi(w_k)] - 2a\gamma m\mathbb{E}[\pi_\phi(w_{k+1})]. \qquad (4.40)
\end{aligned}
$$

Moreover, we have

$$\mathbb{E}[\pi_\phi(w_{k+1})] \leq \xi\mathbb{E}[\pi_\phi(w_k)], \tag{4.41}$$

where

$$\xi = \frac{4\Gamma^2 b^2(M+1) + (1-\mu)^2}{2a\gamma m(1-\mu)^2}. \tag{4.42}$$

Letting $\xi < 1$, we have

$$m \geq \frac{4\Gamma^2 b^2(M+1) + (1-\mu)^2}{2a\gamma(1-\mu)^2} > 0. \tag{4.43}$$

□

Thus, our algorithm SFRCGVR converges linearly for a sufficiently large m.

5. Numerical Examples

Here, we present the comparison of our algorithms SFRCGVR and SPRCGVR with SGD, CG,SVRG, SLBFGS which were implemented in C++ along with the armadillo linear algebra library[29] and Intel MKL. In the experiment, our proposed algorithms SFRCGVR/SPRCGVR show a better performance on some of the well known learning models, which might be indeed non-differentiable convex or non convex models. Ridge regression, logistic regression,regularized hinge and sqhinge losses are some of the widely used learning models that we considered in the implementation of our algorithm for a binary classification of five large scale datasets described in Tab. 1 from the LIBSVM website. In the preprocessing stage, we scaled up each feature values in the range of [-1,+1] for all dimensions using the max-min scaler. The entire dataset were used to minimize the function values of the four learning models for the convergence of our algorithm. For the generalization of our algorithm, we randomly consider one-third of the entire dataset for testing, one-fifth for validation and the remaining for training, that can be used for all algorithms. Having sought the optimal parameter from the candidate set, we set the best trained model using the optimal parameter selected to estimate the AUC scores on the test set.In the implementation of our algorithm, we estimate the gradient $\nabla f(x)$ and

https://software.intel.com/en-us/mkl
https://www.csie.ntu.edu.tw/cjlin/libsvmtools/data sets/

TABLE 1. Experimental Datasets

Dataset	n	d
a9a	32,561	123
covtype	581,012	54
ijcnn1	49,990	22
w8a	49,749	300
SUSY	5,000,000	18

Hessian $\nabla^2 f(x)$ with a small constant $\epsilon = 0.0001$ using numerical methods. However, S-LBFGS computes the Hessian and those classical algorithms CG, SGD, SVRG, and CGVR compute the gradient as well.

5.1. **Parameter Selection.** In our algorithms, we basically consider two main parameters in terms of the number of iterations in both the inner loop m and the outer loop T. We considered the five datasets in the experiment so that the AUC measures is reported after 25 outer loops, i.e., $T = 25$. We initialize a uniformly distributed identical random seed vector w_0 over the interval $[0, 1]^d$, and we particularly consider the size of the sample points $|Q_{k,t}|$ to be $\sqrt{n}$ for SFRCGVR, SPRCGVR and SLBFGS, to compute the gradient and the Hessian matrix. In SLBFGS, we set $M = 10$ and $L = 10$ for memory size and the Hessian update interval, respectively. Besides, we also set $\rho = 0.0001$ and $\sigma = 0.1$ for CG, SFRCGVR and SPRCGVR. Due to momentum, SGD dampens oscillations and accelerated in a certain direction, where the coefficient of the momentum is often set to 0.9. Moreover, we particularly set three different constant step sizes $0.001, 0.0001, 0.00001$ for SVRG, SGD and SLBFGS.

In the implementation, we observe that SFRCGVR rapidly close to the optimal value with few inner loops that indeed gradually decreases as m increases. Hence, SFRCGVR is not sensitive to the parameter m that demands a few inner loops to converge rapidly. However, two algorithms SLBFGS and SVRG reduce the losses a little bit as m increases with an appropriate learning rate. The running time for all algorithms obviously increases as m increases but it varies with the learning rate for SLBFGS and SVRG algorithms. Despite the fact that our algorithm SFRCGVR as well as SPRCGVR are comparable with those algorithms considered under a similar parameter m, it would still have a great benefit of time efficiency due to the few iterations it requires in the inner loop to converge rapidly.

5.2. **Results and Discussion.** We make some comparisons for the convergence of the algorithms by setting the number of inner loop iteration $m = 50$ on five large scale datasets. Having set the optimal learning rate from $0.001, 0.0001, 0.00001$ on the validation set, we correspondingly chose the best model along with these optimal learning rates for SGD, SLBFGS and SVRG.

Fig. 1 considers the x-axis as the number of iterations in the outer loop and y-axis to be the logarithmic values of the loss functions. As can be seen in the convergence of the five algorithms with $\gamma = 0.0001$, CG generally converges faster than SGD, SVRG, and SLBFGS but SFRCGVR is the fastest of all these algorithms on almost all of the four models.However, due to the unsuitability of the learning rates, SGD shows instability on the ridge model as there is a large fluctuation in the loss value. Because SVRG and SLBFGS are sensitive to the learning rates, SLBFGS does not perform better than SVRG. Furthermore, we see that our algorithms converge rapidly in sqhinge model compared to their counterparts.

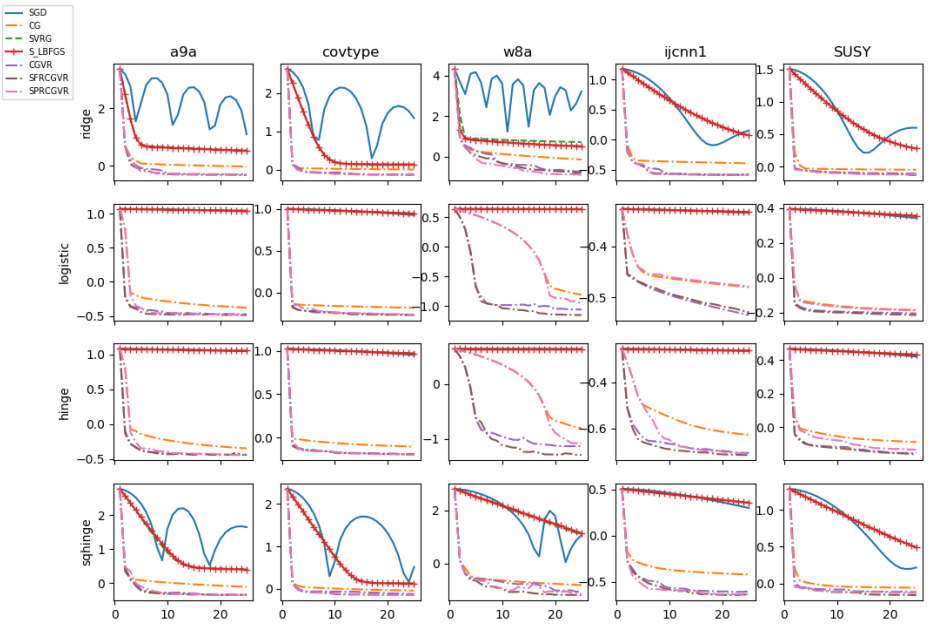

FIGURE 1. Convergence of seven algorithms on the given loss functions with $\gamma = 0.0001$

Fig.2 shows that the average AUC scores of CG, SGD, SLBFGS, SVRG, CGVR, SFR-CGVR and SPRCGVR on five random splits of datasets for each model to make analysis on the generalization of SFRCGVR and SPRCGVR. Overall, either of the algorithms SFR-

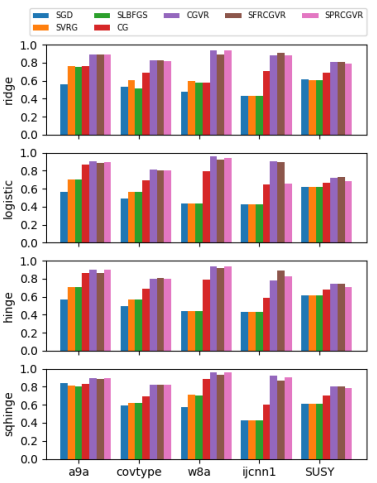

FIGURE 2. AUC scores' comparison on SFRCGVR and SPRCGVR with the corresponding other five algorithms

CGVR or SPRCGVR essentially outperform their counterparts on five large scale data sets. In particular, SLBFGS and SVRG have nearly the same generalization performance but the classical CG algorithm remains outperform these two algorithms. However, the CGVR algorithm outperforms better in general except our SFRCGVR/SPRCGVR algorithms, which have some subtle differences with it. Thus, we can infer that our algorithm SFRCGVR/SPRCGVR with an optimal value achieved has a better generalization performance in general with a suitable regularization settings.

6. Conclusion

This paper presented a new spectral conjugate gradient algorithm based on the reduction of variance with a linear convergence property. The proposed algorithm SFRCGVR/SPRCGVR required a few iteration to converge quickly when compared to SVRG and it needs less memory space while running the algorithm, similar to CG which stores the last gradient vector only, unlike the SLBFGS that stores M vector pairs. Thus, it outperforms and shows better performance on four well known learning models. Moreover, SFRCGVR shows a comparable generalization performance with that of LIBLINEAR solver in optimizing a particular sqhinge loss which results in improving computational efficiency for large scale machine learning problems.

Acknowledgements

This research project is supported by King Mongkut's University of Technology Thonburi (KMUTT), Thailand Science Research and Innovation (TSRI) , and National Science, Research and Innovation Fund (NSRF) Fiscal year 2024 Grant number FRB660073/0164. In addition, this research has received partial partner CaRe Global Network Project funding support from the NSRF via the Program Management Unit for Human Resources & Institutional Development, Research and Innovation [grant number B41G680025].

References

[1] Abbo, K.; Farah, H. Spectral Fletcher-Reeves Algorithm for Solving Non-Linear Unconstrained Optimization Problems. *Iraqi J. Stat. Sci.* **19** (2011), 21–38.

[2] Andrei, N. New accelerated conjugate gradient algorithms as a modification of Dai-Yuan's computational scheme for unconstrained optimization. *J. Comput. Appl. Math.* **234** (2010), no. 12, 3397–3410.

[3] Birgin, G.; Ļosě, M. A spectral conjugate gradient method for unconstrained optimization. *Appl. Math. and optim.* **43** (2001), 117–128.

[4] Bottou, L. Large-scale machine learning with stochastic gradient descent. *Proceedings of COMPSTAT'2010*, 177–186, 2010.

[5] Dai, Y. H.; Yuan, Y. A nonlinear conjugate gradient method with a strong global convergence property. *SIAM J. Optim.* **10** (1999), no. 1, 177–182.

[6] Dennis, J.; John, E. Practical Methods of Optimization: Unconstrained Optimization. *SIAM Review.* **24** (1982), no. 1, 97–98.

[7] Dozat, T. Incorporating nesterov momentum into adam. *Proceedings of the 4th International Conference on Learning Representations*, 1–4, 2016.

[8] Du, S.Q.; Chen, Y.Y. Global convergence of a modified spectral FR conjugate gradient method. *Appl. Math. Comput.* **202** (2008), no. 2, 766–770.

[9] Fletcher, R.; Reeves, C. M. Function minimization by conjugate gradients. *Comput. J.* **7** (1964), 149–154.

[10] Faramarzi, P; Amini, K. A modified spectral conjugate gradient method with global convergence. *J. Optim. Theory Appl.* **182** (2019), no. 2, 667–690.

[11] Gower, R.; Donald, G.; Peter, R. Stochastic block BFGS: Squeezing more curvature out of data. *International Conference on Machine Learning*, 1869–1878, 2016.

[12] Hestenes, M. R.; Eduard, S. Methods of conjugate gradients for solving linear systems. *J. res. National Bur. Stand.* **49** (1952), no. 6, 409–436.

[13] Jian, J.; Lin, Y.; Xianzhen, J.; Pengjie, L.; Meixing, L. A spectral conjugate gradient method with descent property. *Math.* **8** (2020), no. 2, 280.
[14] Jin, X.; Cheng, L.; Xinwen, H. Regularized margin-based conditional log-likelihood loss for prototype learning. *Pattern Rec.* **43** (2010), no. 7, 2428–2438.
[15] Jin, X.B.; Geng, G.G.; Xie, G.S.; Huang, K. Approximately optimizing NDCG using pair-wise loss. *Inform. Sci.* **453** (2018), 50–65.
[16] Jin, X.B.; Geng, G.G.; Sun, M.; Zhang, D. Combination of multiple bipartite ranking for multipartite web content quality evaluation. *Neurocomp.* **149** (2015), 1305–1314.
[17] Jin, X.B.; Guo-Sen, X.; Kaizhu, H.; Amir, H. Accelerating infinite ensemble of clustering by pivot features. *Cogn. Comp.* **10** (2018), 1042–1050.
[18] Jin, X.B.; Zhang, X.Y.; Huang, K.; Geng, G.G. Stochastic conjugate gradient algorithm with variance reduction. *IEEE Trans. Neural Netw. Learn. Syst.* **30** (2019), no. 5, 1360–1369.
[19] Johnson, R.; Tong, Z. Accelerating stochastic gradient descent using predictive variance reduction. *Advances in Neural Inform. Proc. Syst.* **26** (2013).
[20] Kingma, D.P.; Jimmy, B. A. Adam: A method for stochastic optimization. *International Conference on Learning Representations.* (2014).
[21] Kou, C; Yang, H. A mini-batch stochastic conjugate gradient algorithm with variance reduction. *J. Global Optim.* **87** (2023), no. 2-4, 1009–1025.
[22] Liu, J. K.; Feng, Y. M.; Zou, L. M. A spectral conjugate gradient method for solving large-scale unconstrained optimization. *Comput. Math. Appl.* **77** (2019), no. 3, 731–739.
[23] Liu, Y.; Storey, C. Efficient generalized conjugate gradient algorithms, part 1: theory. *J. optim. theory and Appl.* **69** (1991), 129–137.
[24] Mokhtari, A.; Ribeiro, A. RES: regularized stochastic BFGS algorithm. *IEEE Trans. Signal Process.* **62** (2014), no. 23, 6089–6104.
[25] Moritz, P.; Robert, N.; Michael, J. A linearly-convergent stochastic L-BFGS algorithm. In *Artif. Intel. and Stat.* (2016), 249–258.
[26] Polak, E.; Ribière, G. Note sur la convergence de méthodes de directions conjuguées. *(French) Rev. Française Informat. Recherche Opérationnelle* **3** (1969), no. 16, 35–43.
[27] Reddi, S. J.; Satyen, K.; Sanjiv, K. On the convergence of adam and beyond. *arXiv prep.* (2019).
[28] Roux, N.; Mark, S.; Francis, B. A stochastic gradient method with an exponential convergence rate for finite training sets. *Advances in Neural Inform. Processing Syst.* **25** (2012).
[29] Sanderson, C.; Ryan, C. Armadillo: a template-based C++ library for linear algebra. *J. Open Source Soft.* **1** (2016), no. 2, 26.
[30] Shalev-Shwartz, S.; Zhang, T. Stochastic dual coordinate ascent methods for regularized loss minimization. *J. Mach. Learn. Res.* **14** (2013), 567–599.
[31] Sutskever, I.; James, M.; George, D.; Geoffrey, H. On the importance of initialization and momentum in deep learning. In *International conference on machine learning*. (2013), 1139–1147.
[32] Wan, Z.; Jie, G.; Jingjing, L.; Weiyi, L. A modified spectral conjugate gradient projection method for signal recovery. *Signal, Image and Video Processing* **12** (2018), 1455–1462.
[33] Wang, X.; Ma, S.; Goldfarb, D.; Liu, W. Stochastic quasi-Newton methods for nonconvex stochastic optimization. *SIAM J. Optim.* **27** (2017), no. 2, 927–956.
[34] Wu, X. A new spectral Polak-Ribière-Polak conjugate gradient method. *Sci. Asia* **41** (2015), no. 5, 345–349.
[35] Zhang, X.Y.; Wang, L.; Xiang, S.; Liu, C.L. Retargeted least squares regression algorithm. *IEEE Trans. Neural Netw. Learn. Syst.* **26** (2015), no. 9, 2206–2213.

[1] Department of Mathematics, Debre Berhan University, Ethiopia
Email address: seifuendris@gmail.com

[2] Center of Excellence in Theoretical and Computational Science (TaCS-CoE) and Fixed Point Research Laboratory, Room SCL 802 Fixed Point Laboratory, Science Laboratory building, Department of Mathematics, Faculty of Science, King Mongkut's University of Technology Thonburi (KMUTT), 126 Pracha Uthit Rd., Bang Mod, Thung Khru, Bangkok 10140, Thailand
Email address: poom.kum@kmutt.ac.th

[3] NCAO Research Center, Fixed Point Theory and Applications Research Group, Center of Excellence in Theoretical and Computational Science (TaCS-CoE), Faculty of Science, King Mongkut's University of Technology Thonburi (KMUTT), 126 Pracha Uthit Rd., Bang Mod, Thung Khru, Bangkok 10140, Thailand
Email address: parin.cha@mail.kmutt.ac.th