Sample Efficient Decentralized Stochastic Frank-Wolfe Methods for Continuous DR-Submodular Maximization

Hongchang Gao*, **Hanzi Xu** and **Slobodan Vucetic**
Department of Computer and Information Sciences, Temple University, PA, USA
{hongchang.gao, tun47067, vucetic}@temple.edu

Abstract

Continuous DR-submodular maximization is an important machine learning problem, which covers numerous popular applications. With the emergence of large-scale distributed data, developing efficient algorithms for the continuous DR-submodular maximization, such as the decentralized Frank-Wolfe method, became an important challenge. However, existing decentralized Frank-Wolfe methods for this kind of problem have the sample complexity of $\mathcal{O}(1/\epsilon^3)$, incurring a large computational overhead. In this paper, we propose two novel sample efficient decentralized Frank-Wolfe methods to address this challenge. Our theoretical results demonstrate that the sample complexity of the two proposed methods is $\mathcal{O}(1/\epsilon^2)$, which is better than $\mathcal{O}(1/\epsilon^3)$ of the existing methods. As far as we know, this is the first published result achieving such a favorable sample complexity. Extensive experimental results confirm the effectiveness of the proposed methods.

1 Introduction

Continuous DR-submodular maximization, which generalizes the diminishing returns property to the continuous domain, has attracted increasing attention in recent years due to its superior performance on various applications, such as resource allocation [Eghbali and Fazel, 2016; Staib and Jegelka, 2017], learning assignments [Eghbali and Fazel, 2016], and recommendation system [Mokhtari *et al.*, 2018; Xie *et al.*, 2019]. To optimize continuous DR-submodular functions for large-scale data, several stochastic Frank-Wolfe methods [Hassani *et al.*, 2019; Mokhtari *et al.*, 2020; Zhang *et al.*, 2020; Hassani *et al.*, 2017] have been proposed. However, with the emergence of large-scale *distributed* data, maximizing continuous DR-submodular functions with good computational time guarantees becomes a practically relevant open challenge. For example, for the data generated on mobile devices or monitoring sensors, it is prohibitive to upload the data to a central server to conduct centralized optimization due to communication and privacy concerns. Therefore, in this paper we study the continuous DR-submodular maximization in the decentralized scenario. Specifically, we are interested in optimizing the following function:

$$\max_{\mathbf{x}\in\Omega} f(\mathbf{x}) \triangleq \frac{1}{K}\sum_{k=1}^{K} f^{(k)}(\mathbf{x}) \ , \tag{1}$$

where k is the index for workers, $\mathbf{x} \in \Omega$ denotes the model parameters in the compact convex set Ω, $f^{(k)}(\mathbf{x}) = \mathbb{E}_{\xi\sim\mathcal{D}^{(k)}} F^{(k)}(\mathbf{x};\xi)$ is the monotone continuous DR-submodular function on the k-th worker where $F^{(k)}$ is the local cost and $\mathcal{D}^{(k)}$ is the local data distribution. The goal of Eq. (1) is to collaboratively learn the model parameters $\mathbf{x} \in \Omega$ using the data from K workers. In this process, data are kept on each worker and only the model parameters (or gradients) are communicated across different workers.

Decentralized optimization methods, such as decentralized stochastic gradient descent (D-SGD), have been commonly used for training machine learning models when data are distributed on different devices. For instance, [Lian *et al.*, 2017] studied the convergence rate of D-SGD for optimizing non-convex problems. [Koloskova *et al.*, 2019] developed a communication-efficient D-SGD method to reduce the communication cost. [Pu and Nedić, 2020] proposed a gradient tracking technique to improve the convergence performance of D-SGD. However, all of these methods focus on the regular unconstrained machine learning problems. They are not applicable to solve the constrained optimization problems defined in Eq. (1).

To facilitate continuous DR-submodular maximization, several papers [Mokhtari *et al.*, 2018; Xie *et al.*, 2019; Wai *et al.*, 2017] propose decentralized variants of Frank-Wolfe method. In those variants, each worker computes the stochastic gradient based on its local data and then communicates the gradient with its neighbors. Based on the received stochastic gradients from its neighbors, each worker optimizes its local cost function with Frank-Wolfe method. For instance, based on the gossip mechanism, [Mokhtari *et al.*, 2018] proposed a decentralized stochastic Frank-Wolfe (DeSCG) method for optimizing Eq. (1). To achieve ϵ-accuracy tight approximation ratio, it requires $\mathcal{O}(1/\epsilon^3)$ sample complexity and $\mathcal{O}(1/\epsilon^3)$ communication complexity. Based on the gradient tracking technique, [Xie *et al.*, 2019] proposed the decentralized stochastic gradient tracking Frank-Wolfe

*Corresponding author

(DeSGTFW) method, improving the communication complexity to $\mathcal{O}(1/\epsilon)$. However, DeSGTFW still has the same sample complexity as DeSCG because it has to use $\mathcal{O}(1/\epsilon^2)$ samples to compute the stochastic gradient at each iteration. It can be seen that both DeSCG and DeSGTFW have a large sample complexity $\mathcal{O}(1/\epsilon^3)$, resulting in a large computation overhead. Then, a natural question follows: *Is it possible to develop a new decentralized optimization method for continuous DR-submodular maximization defined in Eq. (1), with sample complexity comparable to the centralized scenario?*

To answer the aforementioned question, in this paper, we first propose a gossip-based decentralized stochastic variance-reduced Frank-Wolfe (DeSVRFW-gp) method. In contrast to the existing methods [Xie *et al.*, 2019], which have large sample complexity caused by the stochastic gradient application at each iteration, DeSVRFW-gp employs a variance-reduced stochastic gradient to improve the sample complexity. Our theoretical results show that DeSVRFW-gp achieves $\mathcal{O}(1/\epsilon^2)$ sample complexity, which is better than the previous work. However, DeSVRFW-gp can only converge to the neighborhood of the approximated solution. The reason is that the gossip communication strategy leads to a loose consensus error bound for the gradient. Therefore, to address this issue, we further propose a gradient-tracking-based decentralized stochastic variance-reduced Frank-Wolfe (DeSVRFW-gt) method where we combine the variance reduction technique and gradient tracking strategy. As a result, DeSVRFW-gt can get a tighter consensus error bound for the gradient than DeSVRFW-gp, and then it can asymptotically converge to the approximated solution. Moreover, the theoretical result demonstrates that DeSVRFW-gt also achieves $\mathcal{O}(1/\epsilon^2)$ sample complexity and enjoys the same communication complexity $\mathcal{O}(1/\epsilon)$ as DeSVRFW-gp. The comparison between our methods and the existing methods is summarized in Table 1. In addition to the theoretical considerations, we perform extensive experimental evaluation to confirm the effectiveness of our proposed methods.

To the best of our knowledge, this is the first work applying the variance-reduced stochastic gradient to decentralized Frank-Wolfe methods. The variance reduction technique introduces several challenges when bounding the consensus error by the convergence analysis of the decentralized Frank-Wolfe method. Additionally, the consensus error among different workers brings another challenge when bounding the gradient variance. Our work is the first attempt we are familiar with at providing the convergence rate under this setting. The contributions of this paper are summarized as follows:

- We propose two novel decentralized stochastic variance-reduced Frank-Wolfe methods, namely, DeSVRFW-gp and DeSVRFW-gt. Both of them achieve $\mathcal{O}(1/\epsilon^2)$ sample complexity and $\mathcal{O}(1/\epsilon)$ communication complexity.
- We provide novel theoretical analysis for the proposed methods. Especially, we demonstrate how to bound the *gradient consensus error* for the decentralized stochastic variance-reduced Frank-Wolfe method.
- The extensive empirical results confirm the effectiveness of the proposed DeSVRFW-gp and DeSVRFW-gt.

Methods	Sample	Communication
DeSCG	$\mathcal{O}(1/\epsilon^3)$	$\mathcal{O}(1/\epsilon^3)$
DeSGTFW	$\mathcal{O}(1/\epsilon^3)$	$\mathcal{O}(1/\epsilon)$
DeSVRFW-gp (Ours)	$\mathcal{O}(1/\epsilon^2)$	$\mathcal{O}(1/\epsilon)$
DeSVRFW-gt (Ours)	$\mathcal{O}(1/\epsilon^2)$	$\mathcal{O}(1/\epsilon)$

Table 1: Sample and communication complexities of different methods.

2 Related Work

In this section, we give an overview of the existing work on the continuous DR-submodular maximization.

Continuous DR-submodular maximization. Submodular functions have a diminishing return property. They have been applied to a wide variety of machine learning applications, such as sensor placement [Krause *et al.*, 2006] and human brain network analysis [Salehi *et al.*, 2017]. The continuous DR-submodular function tailors the diminishing return to the continuous domain for addressing some new challenges in machine learning, such as budget allocation [Staib and Jegelka, 2017] and experimental design [Chen *et al.*, 2018]. In this process, optimizing the continuous DR-submodular functions becomes a practically important problem. A number of novel optimization models for this problem were proposed during the last decade. For instance, [Bian *et al.*, 2017; Hassani *et al.*, 2017] showed that the first-order methods can be used for optimizing it approximately. In particular, [Bian *et al.*, 2017] maximized the continuous DR-submodular function with a full conditional gradient descent method approximately and achieves $(1-1/e)\mathrm{OPT}-\epsilon$[1] approximation ratio with the sample complexity of $\mathcal{O}(1/\epsilon)$. Under the stochastic setting, [Hassani *et al.*, 2017] applied the stochastic proximal gradient method to optimize this problem, but it can only achieve $(1/2)\mathrm{OPT}-\epsilon$ approximation ratio and the sample complexity is $\mathcal{O}(1/\epsilon^2)$. [Mokhtari *et al.*, 2020] further proposed a variant of stochastic Frank-Wolfe method, which can achieve $(1-1/e)\mathrm{OPT}-\epsilon$ approximation ratio as the full-gradient based method, but it needs the $\mathcal{O}(1/\epsilon^3)$ sample complexity. Recently, inspired by the development of the variance reduction techniques [Fang *et al.*, 2018; Cutkosky and Orabona, 2019] for stochastic gradients, two variance-reduced stochastic Frank-Wolfe methods [Hassani *et al.*, 2019; Zhang *et al.*, 2020] were proposed. Both of them enjoy $(1-1/e)\mathrm{OPT}-\epsilon$ approximation ratio with only $\mathcal{O}(1/\epsilon^2)$ sample complexity. However, all of these methods only focus on the single machine scenario. It is not clear whether they can retain their favorable complexity when being applied to distributed data.

Decentralized optimization. Decentralized optimization [Lian *et al.*, 2017; Pu and Nedić, 2020; Sun *et al.*, 2019; Yu *et al.*, 2019; Lu *et al.*, 2019; Koloskova *et al.*, 2020; Gao and Huang, 2020] has attracted increasing attention in recent years due to the emergence of decentralized data produced by a variety of smart devices and sensors. Unlike the centralized optimization scenario, there is no central server in

[1] OPT means the optimal function value, e is the natural number.

a decentralized scenario. Each worker only communicates the model parameters or gradients with its neighbors. Thus, there is no communication bottleneck issue associated with the central server. Recently, [Lian *et al.*, 2017] studied the convergence rate of the decentralized stochastic gradient descent (D-SGD) method. They showed that D-SGD enjoys the same convergence rate as the centralized distributed SGD method and the topology of the decentralized system only affects the high-order term of the convergence rate. [Pu and Nedić, 2020] further developed the gradient tracking technique to accelerate the convergence of SGD. In particular, each worker introduces an auxiliary variable to track the average gradients. Recently, [Sun *et al.*, 2019] and [Xin *et al.*, 2020] proposed a new variant based on the variance-reduced gradient, achieving a better communication complexity. However, all these methods only focus on the unconstrained problem and cannot be applied to the constrained optimization problem in Eq. (1).

To facilitate the optimization of Eq. (1) in the decentralized manner, [Mokhtari *et al.*, 2018] developed a gossip-like stochastic Frank-Wolfe method. Specifically, each worker uses the stochastic Frank-Wolfe method to update the local model parameter and then communicates the model parameter and gradient with its neighboring workers. However, this method can only achieve $\mathcal{O}(1/\epsilon^3)$ sample complexity and $\mathcal{O}(1/\epsilon^3)$ communication complexity. To improve it, [Xie *et al.*, 2019] proposed a new decentralized stochastic Frank-Wolfe method based on the gradient tracking technique. The communication complexity is improved to $\mathcal{O}(1/\epsilon)$. However, this method still has a worse sample complexity than that of the single machine methods. Thus, there might still be space to further improve the complexity.

3 Preliminaries

In this section, we introduce the necessary background regarding the decentralized continuous DR-submodular maximization problem.

Definition 1. ***(Continuous submodular function)*** *Given $\mathcal{X} = \prod_{i=1}^{d} \mathcal{X}_i \in \mathbb{R}_+^d$ where $\mathcal{X}_i$ denotes a compact subset of $\mathbb{R}_+$, for the continuous function $F : \mathcal{X} \to \mathbb{R}_+$, if for all $\mathbf{x}, \mathbf{y} \in \mathcal{X}$, we have*

$$F(\mathbf{x}) + F(\mathbf{y}) \geq F(\mathbf{x} \vee \mathbf{y}) + F(\mathbf{x} \wedge \mathbf{y}) \ , \qquad (2)$$

F is submodular. Here, $\mathbf{x} \vee \mathbf{y}$ denotes the element-wise maximization and $\mathbf{x} \wedge \mathbf{y}$ represents the element-wise minimization.

Definition 2. ***(Continuous DR-submodular function)*** *The differentiable submodular function F is called DR-submodular, if for all $\mathbf{x}, \mathbf{y} \in \mathcal{X}$ and $\mathbf{x} \leq \mathbf{y}$[2], it satisfies*

$$\nabla F(\mathbf{x}) \geq \nabla F(\mathbf{y}) \ . \qquad (3)$$

Additionally, a submodular function is called monotone if $F(\mathbf{x}) \geq F(\mathbf{y})$ for $\mathbf{x} \leq \mathbf{y} \in \mathcal{X}$. In this work, we will consider the decentralized optimization of the monotone continuous DR-submodular maximization problem which has a convex constraint Ω, just as shown in Eq. (1).

In the decentralized optimization system, each worker connects with its neighbors, composing a communication network. Here, we use graph $\mathcal{G} = (V, W)$ to represent this communication network. $V = [K]$ denotes all the workers in this system and $W = [w_{ij}] \in \mathbb{R}_+^{K \times K}$ represents the connections among the workers. If the i-th worker and the j-th worker are connected, $w_{ij} > 0$. Otherwise, $w_{ij} = 0$. In addition, the connection matrix W satisfies the following assumption.

Assumption 1. *$W \in \mathbb{R}_+^{K \times K}$ is symmetric ($W^T = W$) and doubly stochastic ($W\mathbf{1} = \mathbf{1}$ and $\mathbf{1}^T W = \mathbf{1}^T$). Regarding the eigenvalues $|\lambda_n| \leq \cdots \leq |\lambda_2| < |\lambda_1| = 1$ of W, they satisfy*

$$\|W - \frac{1}{K}\mathbf{1}\mathbf{1}^T\|_2 \leq 1 - \rho \ , \qquad (4)$$

where $1 - \rho = |\lambda_2| < 1$. Here, $\rho \in (0, 1]$ is called the spectral gap of W.

Assumption 2. *For the compact convex set Ω, its diameter $\sup_{\mathbf{x},\mathbf{y} \in \Omega} \|\mathbf{x} - \mathbf{y}\|$ is bounded by D and its radius $\sup_{\mathbf{x} \in \Omega} \|\mathbf{x}\|$ is bounded by R.*

Assumption 3. *The local objective function on each worker $f^{(k)}(\mathbf{x})$ is L-smooth, i.e.,*

$$\|\nabla f^{(k)}(\mathbf{x}) - \nabla f^{(k)}(\mathbf{y})\| \leq L\|\mathbf{x} - \mathbf{y}\|, \quad \forall x \in \mathcal{X}, \forall y \in \mathcal{X}, \forall k \ , \qquad (5)$$

where $L > 0$ is a constant.

Assumption 4. *The stochastic gradient of the local objective function on each worker is bounded, i.e.,*

$$\mathbb{E}[\|\nabla F^{(k)}(\mathbf{x}; \xi)\|] \leq G, \quad \forall \mathbf{x} \in \mathcal{X}, \forall k \ , \qquad (6)$$

where ξ denotes the randomly selected sample and $G > 0$ is a constant.

Assumption 5. *The variance of stochastic gradients on each worker is bounded, i.e.*

$$\mathbb{E}[\|\nabla F^{(k)}(\mathbf{x}; \xi) - \nabla f^{(k)}(\mathbf{x})\|^2] \leq \sigma^2, \quad \forall \mathbf{x} \in \mathcal{X}, \forall k \ , \qquad (7)$$

where ξ denotes the randomly selected sample and $\sigma > 0$ is a constant.

Algorithm 1 DeSVRFW-gp

```
Initialization: $\mathbf{x}_1^{(k)} = \mathbf{x}_1, T$.
for $t = 1, 2, \cdots, T$ do
    if $t = 1$ then
        Draw $S_1^{(k)}$ samples and compute
            $\mathbf{v}_t^{(k)} = \nabla F^{(k)}(\mathbf{x}_t^{(k)}; S_1^{(k)})$
    else
        Draw $S_{2,t}^{(k)}$ samples and compute
            $\mathbf{v}_t^{(k)} = \mathbf{v}_{t-1}^{(k)} + \nabla F^{(k)}(\mathbf{x}_t^{(k)}; S_{2,t}^{(k)}) - \nabla F^{(k)}(\mathbf{x}_{t-1}^{(k)}; S_{2,t}^{(k)})$
    end if
    $\mathbf{y}_t^{(k)} = \sum_{j \in \mathcal{N}_k} w_{kj} \mathbf{v}_t^{(j)}$
    $\mathbf{u}_t^{(k)} = \arg\max_{\mathbf{u} \in \Omega} \langle \mathbf{y}_t^{(k)}, \mathbf{u} \rangle$
    $\mathbf{x}_{t+1}^{(k)} = \sum_{j \in \mathcal{N}_k} w_{kj} \mathbf{x}_t^{(j)} + \frac{1}{T} \mathbf{u}_t^{(k)}$
end for
```

[2] It means that each coordinate of $\mathbf{x}$ is less than that of $\mathbf{y}$. So does Eq. (3).

4 Decentralized Stochastic Variance-Reduced Frank-Wolfe Method

4.1 Gossip-based Decentralized Stochastic Variance-Reduced Frank-Wolfe Method

In Algorithm 1, we propose the decentralized stochastic variance-reduced Frank-Wolfe method based on the gossip communication strategy (DeSVRFW-gp). At the t-th iteration, the k-th worker computes the variance-reduced gradient $\mathbf{v}_t^{(k)}$ based on its local data, and then communicates $\mathbf{v}_t^{(k)}$ with its neighbors by using the gossip algorithm as follows:

$$\mathbf{y}_t^{(k)} = \sum_{j\in\mathcal{N}_k} w_{kj}\mathbf{v}_t^{(j)} \ , \tag{8}$$

where $\mathcal{N}_k$ denotes the neighboring workers of the k-th worker. It can be seen that $\mathbf{y}_t^{(k)}$ is the weighted average of the neighboring $\mathbf{v}_t^{(j)}$ after the gossip communication. With $\mathbf{y}_t^{(k)}$, the k-th worker optimizes the following linear programming problem to get the feasible ascent direction as the regular Frank-Wolfe method:

$$\mathbf{u}_t^{(k)} = \arg\max_{\mathbf{u}\in\Omega}\langle\mathbf{y}_t^{(k)}, \mathbf{u}\rangle \ . \tag{9}$$

After that, it updates the local model parameters as follows:

$$\mathbf{x}_{t+1}^{(k)} = \sum_{j\in\mathcal{N}_k} w_{kj}\mathbf{x}_t^{(j)} + \frac{1}{T}\mathbf{u}_t^{(k)} \ , \tag{10}$$

where the first term on the right-hand side is to get the model parameter from its neighboring workers by using the gossip algorithm, and the second term is to update the local model parameter with the feasible ascent direction $\mathbf{u}_t^{(k)}$.

In Algorithm 1, we use the variance-reduced technique to reduce the variance of local stochastic gradients. On the contrary, existing works, such as DeSGTFW [Xie *et al.*, 2019], just use the stochastic gradient as $\mathbf{v}_t^{(k)}$. As a result, DeSGTFW has to use a large batch size of samples to reduce the gradient variance, resulting in a large sample complexity at each iteration. Here, inspired by [Fang *et al.*, 2018], we use the variance-reduced gradient to improve the sample complexity. As shown in Algorithm 1, at the first iteration, the k-th worker randomly selects $S_1^{(k)}$ samples and computes the stochastic gradient $\mathbf{v}_1^{(k)}$. At other iterations, each worker selects $S_{2,t}^{(k)}$ samples and computes the following variance-reduced gradient estimator:

$$\mathbf{v}_t^{(k)} = \mathbf{v}_{t-1}^{(k)} + \nabla F^{(k)}(\mathbf{x}_t^{(k)}; S_{2,t}^{(k)}) - \nabla F^{(k)}(\mathbf{x}_{t-1}^{(k)}; S_{2,t}^{(k)}) \ . \tag{11}$$

This gradient estimator has a smaller variance compared with the standard stochastic gradient [Fang *et al.*, 2018]. As a result, it does not need to sample a large batch of samples to reduce the gradient variance as [Xie *et al.*, 2019]. Thus, our method is supposed to have a smaller sample complexity than existing works. In particular, we establish the following theorem to demonstrate the complexity of our Algorithm 1.

Theorem 1. *For Algorithm 1, under Assumptions 1-5, when* $|S_1^{(k)}| = \frac{\sigma^2T^2}{L^2D^2}$ *and* $|S_{2,t}^{(k)}| = \frac{9R^2T}{\rho^2D^2}$ *for* $\forall k, t$, *we have*

$$\begin{aligned} f(\overline{\mathbf{x}}_{T+1}) \geq (1-\frac{1}{e})f(\mathbf{x}^*) - \frac{DRL}{\rho T} - \frac{LR^2}{2T} - \frac{\sqrt{2}LD^2}{T} \\ - \frac{3(1-\rho)LDR}{2\rho} - (1-\rho)DG \ . \end{aligned} \tag{12}$$

Remark 1. *From Theorem 1, it can be seen that DeSVRFW-gp needs* $O(1/\epsilon)$ *iterations to achieve the* ϵ*-accuracy approximation ratio when optimizing Eq. (1). It also indicates that the communication complexity of our method is* $\mathcal{O}(1/\epsilon)$*, which can match the best existing result [Xie* et al.*, 2019]. Thus, our method is efficient in communication.*

Remark 2. *Since the number of iterations* T *is* $O(1/\epsilon)$*, the size of* $S_1^{(k)}$ *is* $O(1/\epsilon^2)$ *and that of* $S_{2,t}^{(k)}$ *is* $O(1/\epsilon)$*. Then, the sample complexity of Algorithm 1 is* $|S_1^{(k)}| + |S_{2,t}^{(k)}| * T = O(1/\epsilon^2)$*. On the contrary, existing methods, such as DeSCG [Mokhtari* et al.*, 2018] and DeSGTFW [Xie* et al.*, 2019], require* $\mathcal{O}(1/\epsilon^3)$ *gradient evaluations. Thus, our method is computationally efficient.*

From Eq. (12), it can be seen that the last two terms on the right-hand side are independent on T. Thus, it can only converge to the neighborhood of $(1-\frac{1}{e})f(\mathbf{x}^*)$, which is not satisfactory. To address this issue, in the following subsection, we propose a new method to have a better convergence result.

Algorithm 2 DeSVRFW-gt

```
Initialization: $\mathbf{x}_1^{(k)} = \mathbf{x}_1, T$.
for $t = 1 \cdots, T-1$ do
    if $t = 1$ then
        Draw $S_1^{(k)}$ samples and compute
        $\mathbf{z}_1^{(k)} = \mathbf{v}_1^{(k)} = \nabla F^{(k)}(\mathbf{x}_1^{(k)}; S_1^{(k)})$
    else
        Draw $S_{2,t}^{(k)}$ samples and compute
        $\mathbf{v}_t^{(k)} = \mathbf{v}_{t-1}^{(k)} + \nabla F^{(k)}(\mathbf{x}_t^{(k)}; S_{2,t}^{(k)}) - \nabla F^{(k)}(\mathbf{x}_{t-1}^{(k)}; S_{2,t}^{(k)})$
        $\mathbf{z}_t^{(k)} = \mathbf{y}_{t-1}^{(k)} + \mathbf{v}_t^{(k)} - \mathbf{v}_{t-1}^{(k)}$
    end if
    $\mathbf{y}_t^{(k)} = \sum_{j\in\mathcal{N}_k} w_{kj}\mathbf{z}_t^{(j)}$
    $\mathbf{u}_t^{(k)} = \arg\max_{\mathbf{u}\in\Omega}\langle\mathbf{y}_t^{(k)}, \mathbf{u}\rangle$
    $\mathbf{x}_{t+1}^{(k)} = \sum_{j\in\mathcal{N}_k} w_{kj}\mathbf{x}_t^{(j)} + \frac{1}{T}\mathbf{u}_t^{(k)}$
end for
```

4.2 Gradient-Tracking-based Decentralized Stochastic Variance-Reduced Frank-Wolfe Method

In Algorithm 1, we use the gossip strategy to communicate with neighboring workers. In this subsection, we proposed a new method based on the gradient tracking technique. Specifically, our proposed gradient-tracking-based stochastic variance-reduced Frank-Wolfe method (DeSVRFW-gt) is shown in Algorithm 2. Its high-level idea is similar to Algorithm 1. The difference lies in the strategy of updating gradients. In detail, at the t-th ($t > 1$) iteration, after obtaining the local variance-reduced gradient $\mathbf{v}_t^{(k)}$, each worker k uses the gradient tracking method to track the average of gradients as follows:

$$\mathbf{y}_t^{(k)} = \sum_{j \in \mathcal{N}_k} w_{kj} (\mathbf{y}_{t-1}^{(j)} + \mathbf{v}_t^{(j)} - \mathbf{v}_{t-1}^{(j)}) \, . \tag{13}$$

With this strategy, $\mathbf{y}_t^{(k)}$ is capable of tracking $\frac{1}{K}\sum_{k=1}^{K} \nabla f^{(k)}(\mathbf{x}_t^{(k)})$. The reason is that $\mathbf{y}_t^{(k)}$ is close to $\bar{\mathbf{y}}_t = \frac{1}{K}\sum_{k=1}^{K} \mathbf{y}_t^{(k)}$ and $\bar{\mathbf{y}}_t$ is close to $\frac{1}{K}\sum_{k=1}^{K} \nabla f^{(k)}(\mathbf{x}_t^{(k)})$. Compared with Algorithm 1 which uses the gossip communication method, $\mathbf{y}_t^{(k)}$ can approximate $\bar{\mathbf{y}}_t = \frac{1}{K}\sum_{k=1}^{K} \mathbf{y}_t^{(k)}$ more tightly. In other words, Algorithm 2 has a tighter consensus error $\|\mathbf{y}_t^{(k)} - \bar{\mathbf{y}}_t\|^2$ than Algorithm 1, which will be shown in the next section.

To see the sample and communication complexities of DeSVRFW-gt, we establish the following theorem.

Theorem 2. *For Algorithm 2, under Assumptions 1-5, when $|S_1^{(k)}| = \frac{\sigma^2 T^2}{L^2 D^2}$ and $|S_{2,t}^{(k)}| = \frac{9R^2 T}{\rho^2 D^2}$ for $\forall k, t$, we have*

$$f(\overline{\mathbf{x}}_{t+1}) \geq (1 - \frac{1}{e}) f(\mathbf{x}^*) - \frac{LDR}{\rho T} - \frac{\sqrt{2}LD^2}{T} - \frac{DG}{\rho T} - \frac{\sqrt{12}LD^2 + \sqrt{27}LDR}{\rho^2 T} - \frac{LR^2}{2T} \, . \tag{14}$$

Remark 3. *From Theorem 2, it can be seen that Algorithm 2 can converge to $(1 - \frac{1}{e}) f(\mathbf{x}^*)$ rather than its neighborhood. Thus, Algorithm 2 has a better convergence performance than Algorithm 1.*

Remark 4. *Theorem 2 indicates that the communication complexity of Algorithm 2 is $\mathcal{O}(1/\epsilon)$ to achieve the ϵ-accuracy approximation ratio for optimizing Eq. (1), which is the same as that of Algorithm 1 and can match the best existing result in [Xie* et al.*, 2019].*

Remark 5. *As for the sample complexity of Algorithm 2, it is $\mathcal{O}(1/\epsilon^2)$, which is the same as that of Algorithm 1 and better than DeSCG [Mokhtari* et al.*, 2018] and DeSGTFW [Xie* et al.*, 2019].*

From Theorems 1 and 2, we can find that the batch size $|S_{2,t}^{(k)}|$ at each iteration is $\mathcal{O}(1/\epsilon)$. Compared with DeSGTFW [Xie *et al.*, 2019] whose batch size is $\mathcal{O}(1/\epsilon^2)$, our two methods have a smaller sample complexity at each iteration, accelerating the convergence speed.

In summary, we proposed two computationally efficient decentralized Frank-Wolfe methods to optimize Eq. (1). To the best of our knowledge, this is the first work applying the variance reduction technique to the decentralized Frank-Wolfe method. Although the variance reduction technique has been studied for the single machine case, our methods are novel and challenging, especially for the convergence analysis. On one hand, unlike the traditional variance-reduced Frank-Wolfe methods, our methods need to deal with the consensus error among different workers. On the other hand, the introduced variance-reduced gradient makes it more challenging to handle the consensus error compared with the traditional decentralized Frank-Wolfe methods.

5 Main Proof

In this section, we present the high-level idea of proving the convergence rate of our proposed DeSVRFW-gp and DeSVRFW-gt.

According to the smoothness of the loss function, we can get the following inequality:

$$\begin{aligned} f(\mathbf{x}^*) - f(\overline{\mathbf{x}}_{t+1}) \leq & \frac{1}{e}(f(\mathbf{x}^*) - f(\overline{\mathbf{x}}_1)) + \frac{LR^2}{2T} \\ & + \frac{D}{T}\sum_{t=1}^{T} \underbrace{\|\frac{1}{K}\sum_{k=1}^{K} \nabla f^{(k)}(\mathbf{x}_t^{(k)}) - \overline{\mathbf{y}}_t\|}_{T_1} \\ & + \frac{LD}{T\sqrt{K}}\sum_{t=1}^{T} \underbrace{\|\overline{X}_t - X_t\|_F}_{T_2} + \frac{D}{T\sqrt{K}}\sum_{t=1}^{T} \underbrace{\|\overline{Y}_t - Y_t\|_F}_{T_3} \, . \end{aligned} \tag{15}$$

where T_1 measures how the averaging stochastic gradient $\overline{\mathbf{y}}_t$ is close to the averaging full gradient $\frac{1}{K}\sum_{k=1}^{K} \nabla f^{(k)}(\mathbf{x}_t^{(k)})$, T_2 measures the consensus error regarding the local model parameter, and T_3 measures the consensus error regarding the local stochastic gradient. Thus, to establish the convergence results in Theorem 1 and 2, what we need to do is to bound these three terms.

Due to the space limitation, we only show the bound of the consensus error T_3 for these two theorems. The bounds for T_1 and T_2 can be found in the Supplementary Materials.

Lemma 1. *For Algorithm 1, under Assumption 1-5, when $|S_1| = \frac{\sigma^2 T^2}{L^2 D^2}$ and $|S_2| = \frac{9R^2 T}{\rho^2 D^2}$, we have*

$$\mathbb{E}[\|Y_t - \overline{Y}_t\|_F] \leq (1 - \rho)\sqrt{K}G + \frac{3(1-\rho)\sqrt{K}LR(t-1)}{\rho T} \, . \tag{16}$$

Lemma 2. *For Algorithm 2, under Assumption 1-5, when $|S_1| = \frac{\sigma^2 T^2}{L^2 D^2}$ and $|S_2| = \frac{9R^2 T}{\rho^2 D^2}$, we have*

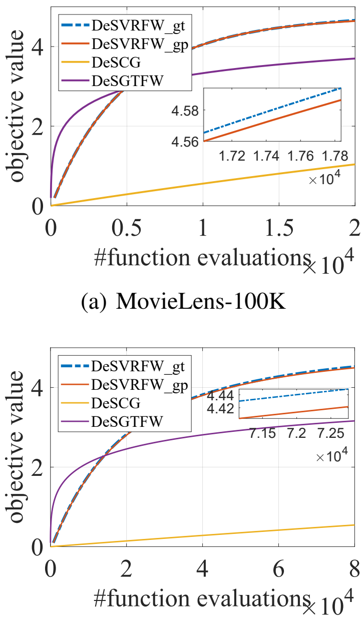
$$\mathbb{E}[\|Y_t - \overline{Y}_t\|_F] \leq (1 - \rho)^{t-1}\sqrt{K}G + \frac{\sqrt{12K}DL + \sqrt{27K}RL}{\rho^2 T} \, . \tag{17}$$

From the above two lemmas, it can be seen that Algorithm 2 can obtain a tighter consensus error $\|Y_t - \overline{Y}_t\|_F$. Specifically, the first term in Lemma 1 is a constant, resulting in the constant term in Theorem 1. Thus, Algorithm 1 can only converge to the neighborhood of $(1 - \frac{1}{e}) f(\mathbf{x}^*)$. On the contrary, all items in Lemma 2 are shrunk when increasing the number of iterations. Thus, its bound is tighter than Lemma 1.

6 Experiments

In this section, we describe the experimental design and results of an empirical evaluation of the proposed methods.

In the experiment, following [Xie *et al.*, 2019; Mokhtari *et al.*, 2018], we apply our proposed DeSVRFW-gp and DeSVRFW-gt to the movie recommendation system. The loss function can be found in Supplementary Material. Here, we use two datasets: MovieLens-1M and MovieLens-100K [3]. In detail, MovieLens-1M has 1 million ratings which are from 6,040 users on 3,883 movies, and MovieLens-100K has 100,000 ratings which are from 943 users on 1,682 movies. Ratings range from 1 to 5. The goal of this experiment is to recommend a group of $|O| = 10$ movies to users expecting the highest satisfaction based on users' historical ratings. In our experiment, we use 8 workers for MovieLens-1M and 4 workers for MovieLens-100K. For each case, the ratings from users are divided into all workers evenly. For instance, when there are 8 workers for MovieLens-1M, each worker will have 755 users and their associated historical ratings.



(a) MovieLens-100K

(b) MovieLens-1M

Figure 1: Comparison between different methods. The plots show the objective value versus the total number of function evaluation calculated.

As for the communication graph, we use the Erdos-Renyi random graph in our experiment. The mean vertex degree in the graph is 2. For the non-diagonal entries in the weight matrix W of the communication graph, if vertex i and vertex j are connected, $w_{ij} = 1/(1 + \max(D_i, D_j))$ where D_i denotes the degree of the vertex i. If there is no edge between vertex i and vertex j, w_{ij} equals to 0. For the diagonal entries, $w_{ii} = 1 - \sum_{j \in \mathcal{N}(i)} w_{ij}$. This kind of weight matrix satisfies Assumption 1.

In Figure 1, we compare our proposed two methods with two baseline methods: DeSCG [Mokhtari *et al.*, 2018] and DeSGTFW [Xie *et al.*, 2019]. In particular, we plot the objective function value with respect to the number of function evaluations for two datasets, respectively. Here, for our methods, we use the batch size of 100 for MovieLens-100K and 200 for MovieLens-1M. Regarding the other two baseline methods, we set it according to their theoretical results.

From Figure 1, we can see that DeSCG converges much slower than all the other methods. The reason is that it uses the stochastic gradient at each iteration which has a large estimation variance, slowing down the convergence speed. In addition, it can be found that DeSGTFW converges faster than DeSCG but slower than our two methods. Here, DeSGTFW employs the gradient tracking technique so that it is faster than DeSCG. However, our methods employ the variance reduced gradient at each iteration. Thus, our two methods converge faster than DeSGTFW, which is consistent with our theoretical results. Note that, in Figure 1, DeSGTFW converges a little faster than our two methods at the beginning phase. The reason is that DeSGTFW uses $O(1/t^2)$ samples at each iteration where t represents the t-th iteration. Therefore, at the beginning phase, DeSGTFW is faster, but it becomes slower as the training progresses because more samples are needed for each update. Furthermore, we observe that DeSVRFW-gt outperforms DeSVRFW-gp. The reason is that DeSVRFW-gt employs the gradient tracking technique to estimate the global gradient across different workers so that $\mathbf{y}_t^{(k)}$ has a smaller variance compared to DeSVRFW-gp. Thus, DeSVRFW-gt converges faster.

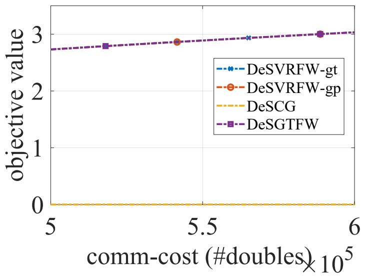


Figure 2: The plots show the objective value versus the communication cost for MovieLens-100K.

Finally, in Figure 2, we plot the objective function value versus the accumulated communication cost. Due to the space limitation, we only show the result for MovieLens-100K. The other dataset produces a similar result. It can be seen that our two methods have almost the same communication cost as DeSGTFW, which is consistent with the theoretical result. DeSCG has the largest communication cost, which is also consistent with Table 1.

7 Conclusions

In this paper, we proposed two novel decentralized stochastic Frank-Wolfe methods for optimizing the continuous DR-Submodular maximization problem. Our theoretical results demonstrated that our two methods have better sample complexities than the existing methods. This is the first work that achieves such sample complexities. Our extensive empirical evaluation is consistent with the theoretically predicted results.

[3] https://grouplens.org/datasets/movielens/

References

[Bian *et al.*, 2017] Andrew An Bian, Baharan Mirzasoleiman, Joachim Buhmann, and Andreas Krause. Guaranteed non-convex optimization: Submodular maximization over continuous domains. In *Artificial Intelligence and Statistics*, pages 111–120. PMLR, 2017.

[Chen *et al.*, 2018] Lin Chen, Hamed Hassani, and Amin Karbasi. Online continuous submodular maximization. *arXiv preprint arXiv:1802.06052*, 2018.

[Cutkosky and Orabona, 2019] Ashok Cutkosky and Francesco Orabona. Momentum-based variance reduction in non-convex sgd. In *Advances in Neural Information Processing Systems*, pages 15236–15245, 2019.

[Eghbali and Fazel, 2016] Reza Eghbali and Maryam Fazel. Designing smoothing functions for improved worst-case competitive ratio in online optimization. In *Advances in Neural Information Processing Systems*, pages 3287–3295, 2016.

[Fang *et al.*, 2018] Cong Fang, Chris Junchi Li, Zhouchen Lin, and Tong Zhang. Spider: Near-optimal non-convex optimization via stochastic path-integrated differential estimator. In *Advances in Neural Information Processing Systems*, pages 689–699, 2018.

[Gao and Huang, 2020] Hongchang Gao and Heng Huang. Periodic stochastic gradient descent with momentum for decentralized training. *arXiv preprint arXiv:2008.10435*, 2020.

[Hassani *et al.*, 2017] Hamed Hassani, Mahdi Soltanolkotabi, and Amin Karbasi. Gradient methods for submodular maximization. In *Advances in Neural Information Processing Systems*, pages 5841–5851, 2017.

[Hassani *et al.*, 2019] Hamed Hassani, Amin Karbasi, Aryan Mokhtari, and Zebang Shen. Stochastic conditional gradient++. *arXiv preprint arXiv:1902.06992*, 2019.

[Koloskova *et al.*, 2019] Anastasia Koloskova, Sebastian U Stich, and Martin Jaggi. Decentralized stochastic optimization and gossip algorithms with compressed communication. *arXiv preprint arXiv:1902.00340*, 2019.

[Koloskova *et al.*, 2020] Anastasia Koloskova, Nicolas Loizou, Sadra Boreiri, Martin Jaggi, and Sebastian U Stich. A unified theory of decentralized sgd with changing topology and local updates. *arXiv preprint arXiv:2003.10422*, 2020.

[Krause *et al.*, 2006] Andreas Krause, Carlos Guestrin, Anupam Gupta, and Jon Kleinberg. Near-optimal sensor placements: Maximizing information while minimizing communication cost. In *Proceedings of the 5th international conference on Information processing in sensor networks*, pages 2–10, 2006.

[Lian *et al.*, 2017] Xiangru Lian, Ce Zhang, Huan Zhang, Cho-Jui Hsieh, Wei Zhang, and Ji Liu. Can decentralized algorithms outperform centralized algorithms? a case study for decentralized parallel stochastic gradient descent. In *Advances in Neural Information Processing Systems*, pages 5330–5340, 2017.

[Lu *et al.*, 2019] Songtao Lu, Xinwei Zhang, Haoran Sun, and Mingyi Hong. Gnsd: a gradient-tracking based nonconvex stochastic algorithm for decentralized optimization. In *2019 IEEE Data Science Workshop, DSW 2019*, pages 315–321. Institute of Electrical and Electronics Engineers Inc., 2019.

[Mokhtari *et al.*, 2018] Aryan Mokhtari, Hamed Hassani, and Amin Karbasi. Decentralized submodular maximization: Bridging discrete and continuous settings. *arXiv preprint arXiv:1802.03825*, 2018.

[Mokhtari *et al.*, 2020] Aryan Mokhtari, Hamed Hassani, and Amin Karbasi. Stochastic conditional gradient methods: From convex minimization to submodular maximization. *Journal of Machine Learning Research*, 21(105):1–49, 2020.

[Pu and Nedić, 2020] Shi Pu and Angelia Nedić. Distributed stochastic gradient tracking methods. *Mathematical Programming*, pages 1–49, 2020.

[Salehi *et al.*, 2017] Mehraveh Salehi, Amin Karbasi, Dustin Scheinost, and R Todd Constable. A submodular approach to create individualized parcellations of the human brain. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 478–485. Springer, 2017.

[Staib and Jegelka, 2017] Matthew Staib and Stefanie Jegelka. Robust budget allocation via continuous submodular functions. In *International Conference on Machine Learning*, pages 3230–3240. PMLR, 2017.

[Sun *et al.*, 2019] Haoran Sun, Songtao Lu, and Mingyi Hong. Improving the sample and communication complexity for decentralized non-convex optimization: A joint gradient estimation and tracking approach. *arXiv preprint arXiv:1910.05857*, 2019.

[Wai *et al.*, 2017] Hoi-To Wai, Jean Lafond, Anna Scaglione, and Eric Moulines. Decentralized frank–wolfe algorithm for convex and nonconvex problems. *IEEE Transactions on Automatic Control*, 62(11):5522–5537, 2017.

[Xie *et al.*, 2019] Jiahao Xie, Chao Zhang, Zebang Shen, Chao Mi, and Hui Qian. Decentralized gradient tracking for continuous dr-submodular maximization. In *The 22nd International Conference on Artificial Intelligence and Statistics*, pages 2897–2906, 2019.

[Xin *et al.*, 2020] Ran Xin, Usman A Khan, and Soummya Kar. A near-optimal stochastic gradient method for decentralized non-convex finite-sum optimization. *arXiv preprint arXiv:2008.07428*, 2020.

[Yu *et al.*, 2019] Hao Yu, Rong Jin, and Sen Yang. On the linear speedup analysis of communication efficient momentum sgd for distributed non-convex optimization. *arXiv preprint arXiv:1905.03817*, 2019.

[Zhang *et al.*, 2020] Mingrui Zhang, Zebang Shen, Aryan Mokhtari, Hamed Hassani, and Amin Karbasi. One sample stochastic frank-wolfe. In *International Conference on Artificial Intelligence and Statistics*, pages 4012–4023. PMLR, 2020.