A Decentralized Proximal Point-type Method for Non-convex Non-concave Saddle Point Problems

Weijie Liu WESTONHUNTER@ZJU.EDU.CN
Zhejiang University
Aryan Mokhtari MOKHTARI@AUSTIN.UTEXAS.EDU
UT Austin
Asuman Ozdaglar ASUMAN@MIT.EDU
Massachusetts Institute of Technology
Sarath Pattathil SARATHP@MIT.EDU
Massachusetts Institute of Technology
Zebang Shen ZEBANG@SEAS.UPENN.EDU
University of Pennsylvania
Nenggan Zheng ZNG@CS.ZJU.EDU.CN
Zhejiang University

Abstract

In this paper, we focus on solving a class of constrained non-convex non-concave saddle point problems in a decentralized manner by a group of nodes in a network. Specifically, we assume that each node has access to a summand of a global objective function and nodes are allowed to exchange information only with their neighboring nodes. We propose a decentralized variant of the proximal point method for solving this problem. We show that when the objective function is ρ-weakly convex-weakly concave the iterates converge to approximate stationarity with a rate of $\mathcal{O}(1/\sqrt{T})$ where the approximation error depends linearly on $\sqrt{\rho}$. We further show that when the objective function satisfies the Minty VI condition (which generalizes the convex-concave case) we obtain convergence to stationarity with a rate of $\mathcal{O}(1/\sqrt{T})$. To the best of our knowledge, our proposed method is the first decentralized algorithm with theoretical guarantees for solving a non-convex non-concave decentralized saddle point problem.

1. Introduction

In this paper we focus on solving the following saddle point problem

$$\min_{\mathbf{x}\in\mathcal{X}} \max_{\mathbf{y}\in\mathcal{Y}} f(\mathbf{x}, \mathbf{y}) \tag{1}$$

where $\mathbf{x} \in \mathcal{X} \subset \mathbb{R}^p$ and $\mathbf{y} \in \mathcal{Y} \subset \mathbb{R}^q$ are the inputs of the function $f : \mathbb{R}^{p\times q} \to \mathbb{R}$. We assume that both sets $\mathcal{X}$ and $\mathcal{Y}$ are nonempty, convex and compact.

We are interested in studying the general non-smooth non-convex non-concave version of Problem (1) in which f could be non-smooth non-convex with respect to the minimization variable $\mathbf{x}$ and non-smooth non-concave with respect to the maximization variable $\mathbf{y}$. Indeed, in such cases, finding an optimal solution of Problem (1) is hard, in general, as we know solving a special case of this problem which is minimizing a non-convex function is hard, in general. Therefore, we settle for finding a set of points $(\mathbf{x}^*, \mathbf{y}^*)$ that satisfy the first-order optimality conditions for Problem (1), i.e.,

$$\left(\nabla_{\mathbf{x}} f(\mathbf{x}^*, \mathbf{y}^*)\right)^\top (\mathbf{x} - \mathbf{x}^*) \geq 0, \qquad \left(\nabla_{\mathbf{y}} f(\mathbf{x}^*, \mathbf{y}^*)\right)^\top (\mathbf{y} - \mathbf{y}^*) \leq 0, \tag{2}$$

for any $\mathbf{x} \in \mathcal{X}$ and $\mathbf{y} \in \mathcal{Y}$. The focus of this paper is on distributed optimization in which the objective function f is defined as a sum of N individual functions $f_n : \mathbb{R}^{p\times q} \to \mathbb{R}$ for $n \in \{1, \ldots, N\}$, i.e., $f(\mathbf{x}, \mathbf{y}) := \sum_{n=1}^{N} f_n(\mathbf{x}, \mathbf{y})$. We assume that each component function f_n is assigned to a node (machine), indexed by n, in a network of size N. We would like to have an algorithm which updates parameters in such a manner that no machine will need access to data from all other machines to update the parameters.

2. Problem Formulation

In this section, we first introduce a formulation of the decentralized version of the general saddle point problem in (1). Consider a network of N nodes (machines) in which each node n has access to a local objective function $f_n : \mathbb{R}^{p\times q} \to \mathbb{R}$. Further, let $\mathbf{x}_n, \mathbf{y}_n$ be the decision variables for node n. Our goal is to solve the following distributed consensus saddle point problem

$$\min_{\{\mathbf{x}_n\}_{n=1}^N} \max_{\{\mathbf{y}_n\}_{n=1}^N} \sum_{n=1}^{N} f_n(\mathbf{x}_n, \mathbf{y}_n) \qquad \text{s.t. } \mathbf{x}_1 = \ldots = \mathbf{x}_N \in \mathcal{X}, \quad \mathbf{y}_1 = \ldots = \mathbf{y}_N \in \mathcal{Y}, \tag{3}$$

where to ensure that nodes find the same stationary point and reach consensus we enforce the iterates of the nodes to be equal to each other, i.e., $\mathbf{x}_1 = \ldots = \mathbf{x}_N$ and $\mathbf{y}_1 = \ldots = \mathbf{y}_N$. It can be easily verified that the distributed problem in (3) is equivalent to the original problem in (1) when we enforce the consensus condition. In other words, a pair $(\mathbf{x}^*, \mathbf{y}^*)$ is an optimal solution of (1) if and only if $\mathbf{x}_1 = \ldots = \mathbf{x}_N = \mathbf{x}^*$ and $\mathbf{y}_1 = \ldots = \mathbf{y}_N = \mathbf{y}^*$ is an optimal solution of (3). Let us define $\mathbf{1}_{\mathcal{X}}$ and $\mathbf{1}_{\mathcal{Y}}$ as the indicator functions of the feasible sets $\mathcal{X}$ and $\mathcal{Y}$, respectively, and use the notation $\partial\mathbf{1}_{\mathcal{X}}$ and $\partial\mathbf{1}_{\mathcal{Y}}$ to denote their corresponding subgradients. Note that $\partial\mathbf{1}_{\mathcal{X}}$ and $\partial\mathbf{1}_{\mathcal{Y}}$ are normal cones of the convex sets $\mathcal{X}$ and $\mathcal{Y}$, respectively, and we can express the conditions in (2) as

$$0 \in \sum_{n=1}^{N} \nabla_{\mathbf{x}} f_n(\mathbf{x}^*, \mathbf{y}^*) + \partial\mathbf{1}_{\mathcal{X}}(\mathbf{x}^*), \qquad 0 \in -\sum_{n=1}^{N} \nabla_{\mathbf{y}} f_n(\mathbf{x}^*, \mathbf{y}^*) + \partial\mathbf{1}_{\mathcal{Y}}(\mathbf{y}^*). \tag{4}$$

Now define the local operators $\mathcal{B}_n : \mathbb{R}^d \to \mathbb{R}^d$ and $\mathcal{R}_n : \mathbb{R}^d \to \mathbb{R}^d$ at node n as $\mathcal{B}_n([\mathbf{x};\mathbf{y}]) = [\nabla_{\mathbf{x}} f_n(\mathbf{x}, \mathbf{y}), -\nabla_{\mathbf{y}} f_n(\mathbf{x}, \mathbf{y})]$ and $\mathcal{R}_n([\mathbf{x};\mathbf{y}]) = [\partial\mathbf{1}_{\mathcal{X}}, \partial\mathbf{1}_{\mathcal{Y}}]$, where to simplify our notation we defined $d := p + q$. Considering these definitions our goal in (2) and (4) can be written as the following decentralized root finding problem

$$\operatorname*{find}_{\{\mathbf{z}_n\}_{n=1}^N} \qquad \text{s.t. } \sum_{n=1}^{N} \mathcal{B}_n(\mathbf{z}_n) + \mathcal{R}_n(\mathbf{z}_n) = 0, \quad \mathbf{z}_1 = \ldots = \mathbf{z}_N \in \mathcal{Z}, \tag{5}$$

where $\mathbf{z}_n \overset{\text{def}}{=} [\mathbf{x}_n; \mathbf{y}_n] \in \mathbb{R}^d$ is the concatenated local variable at node n and the set $\mathcal{Z} \subset \mathbb{R}^d$ is a convex set defined as $\mathcal{Z} := \{[\mathbf{x};\mathbf{y}] \in \mathbb{R}^d \mid \mathbf{x} \in \mathcal{X}, \mathbf{y} \in \mathcal{Y}\}$. To further simplify our problem formulation and write it in a more compact form, let us define the vector $\mathbf{z} = [\mathbf{z}_1; \ldots; \mathbf{z}_N] \in \mathbb{R}^{Nd}$ as the concatenation of all the local variables, $\mathcal{B}(\mathbf{z}) := [\mathcal{B}_1(\mathbf{z}_1); \cdots; \mathcal{B}_N(\mathbf{z}_N)] \in \mathbb{R}^{Nd}$ as the operator corresponding to the aggregate objective function, $\mathcal{R}(\mathbf{z}) := [\mathcal{R}_1(\mathbf{z}_1); \cdots; \mathcal{R}_N(\mathbf{z}_N)] \in \mathbb{R}^{Nd}$ as

Algorithm 1 DPPSP at node n

Input: initial iterate $\mathbf{z}_n^0$, step size α, weights w_{nm} for $m \in \mathcal{N}_n$;
1: **for** $t = 0, \ldots, T-1$ **do**
2: Exchange variable $\mathbf{z}_n^t$ with neighboring nodes $m \in \mathcal{N}_n$;
3: **if** $t = 0$ **then**
4: $\mathbf{z}^{t+1} = (\mathbf{I} + \alpha(\mathcal{B}_n + \mathcal{R}_n))^{-1}\big(\sum_{m\in\mathcal{N}_n}(2w_{nm} - 1)\mathbf{z}_m^t\big);$
5: **else**
6: $\mathbf{z}_n^{t+1} = (\mathbf{I} + \alpha(\mathcal{B}_n + \mathcal{R}_n))^{-1}\big(\sum_{m\in\mathcal{N}_n} w_{nm}(2\mathbf{z}_m^t - \mathbf{z}_m^{t-1}) + \alpha[\mathcal{B}(\mathbf{z}_n^t) + \mathcal{R}(\mathbf{z}_n^t)]\big);$
7: **end if**
8: **end for**

the aggregate operator corresponding to the indicator functions, and $\mathcal{Z}^N \subset \mathbb{R}^{Nd}$ is given by $\mathcal{Z}^N := \{[\mathbf{z}_1; \ldots; \mathbf{z}_N] \in \mathbb{R}^{Nd} \mid \mathbf{z}_1, \ldots, \mathbf{z}_N \in \mathcal{Z}\}$. Then, the problem in (5) can be stated as

$$\operatorname*{find}_{\mathbf{z}} \quad \text{s.t.}\ (\mathbf{1}_N \otimes \mathbf{I}_d)^\top(\mathcal{B}(\mathbf{z}) + \mathcal{R}(\mathbf{z})) = 0, \quad (\mathbf{I}_{Nd} - \hat{\mathbf{W}})\mathbf{z} = \mathbf{0}_{Nd}, \quad \mathbf{z} \in \mathcal{Z}^N, \tag{6}$$

where we replaced the consensus constraint $\mathbf{z}_1 = \ldots = \mathbf{z}_N$ with the condition $(\mathbf{I} - \hat{\mathbf{W}})\mathbf{z} = \mathbf{0}$. Here the matrix $\hat{\mathbf{W}} = \mathbf{W} \otimes \mathbf{I}_d \in \mathbb{R}^{Nd\times Nd}$ is the Kronecker product of the identity matrix $\mathbf{I}_d \in \mathbb{R}^{d\times d}$ and a mixing matrix $\mathbf{W} \in \mathbb{R}^{N\times N}$, where $\mathbf{W}$ has the sparsity pattern of the graph and is designed in a way that the constraint $(\mathbf{I} - \hat{\mathbf{W}})\mathbf{z} = \mathbf{0}$ is satisfied if and only if $\mathbf{z}_1 = \ldots = \mathbf{z}_N$. This is a common approach in decentralized optimization [49], and if $\mathbf{W}$ satisfies the following conditions

$$\mathbf{W} = \mathbf{W}^T, \quad \mathbf{W}\mathbf{1}_N = \mathbf{1}_N, \quad \text{null}(\mathbf{I} - \mathbf{W}) = \text{span}\{\mathbf{1}_N\}, \quad \mathbf{0}_N \prec \mathbf{W} \preceq \mathbf{I}_N, \tag{7}$$

then (5) and (6) are equivalent. The equality conditions in (6) are coupled as $\mathbf{z}$ has to be chosen for both of them at the same time. We proceed to define a new variable $\mathbf{q}$ (which behaves as the dual variable for the consensus constraint) to separate these equality conditions. If we define $\mathbf{U} \triangleq (\mathbf{I} - \hat{\mathbf{W}})^{1/2}$, then the optimality conditions of Problem (6) imply that there exists some $\mathbf{p}^* \in \mathbb{R}^{Nd}$, such that for $\mathbf{q}^* = \mathbf{U}\mathbf{p}^* \in \mathbb{R}^{Nd}$ and $\alpha > 0$ we have

$$\mathbf{U}\mathbf{q}^* + \alpha[\mathcal{B}(\mathbf{z}^*) + \mathcal{R}(\mathbf{z}^*)] = \mathbf{0} \quad \text{and} \quad \mathbf{U}\mathbf{z}^* = \mathbf{0}, \tag{8}$$

where $\mathbf{z}^* \in \mathbb{R}^{Nd}$ is a solution of Problem (6); see, e.g., [40]. Equation (8) is obtained from the first order stationarity conditions of Problem (6) where $\frac{1}{\alpha}\mathbf{p}^*$ can be seen as the Lagrange multiplier for the constraint in (6). The first equation follows from the optimality condition and the second one follows from the constraint, since $-\mathbf{U}\mathbf{z}^* = \mathbf{0} \iff (\mathbf{I} - \hat{\mathbf{W}})\mathbf{z}^* = 0$. Hence, instead of solving (6) we find the optimal pair $(\mathbf{z}^*, \mathbf{q}^*)$ for (8) by updating $\mathbf{z}$ and $\mathbf{q}$ alternatively, as we do in the following section.

3. Algorithm

In this section, we design a decentralized method to solve the root finding problem introduced in (8) which leads to a set of local iterates satisfying the conditions in (4). If we define $\mathbf{v} = [\mathbf{z}; \mathbf{q}] \in \mathbb{R}^{2Nd}$ as the concatenation of $\mathbf{z}$ and $\mathbf{q}$, then our problem is equivalent to finding the root of $\mathcal{T}(\mathbf{v})$, where the operator $\mathcal{T} : \mathbb{R}^{2Nd} \to \mathbb{R}^{2Nd}$ is defined as

$$\mathcal{T}(\mathbf{v}) = \Bigg(\underbrace{\begin{bmatrix} \alpha[\mathcal{B}+\mathcal{R}] & 0 \\ 0 & 0 \end{bmatrix}}_{\mathcal{T}_1} + \underbrace{\begin{bmatrix} \mathbf{0} & \mathbf{U} \\ -\mathbf{U} & \mathbf{0} \end{bmatrix}}_{\mathcal{T}_2} \Bigg) \underbrace{\begin{bmatrix} \mathbf{z} \\ \mathbf{q} \end{bmatrix}}_{\mathbf{v}}, \tag{9}$$

i.e., finding a point $\mathbf{v}^*$ such that $\mathcal{T}(\mathbf{v}^*) = \mathbf{0}$. Under the condition that the operator $(\mathcal{I}+\mathcal{T})^{-1}$ is well-defined, finding a root of $\mathcal{T}$ is equivalent to finding a fixed point of the operator $(\mathcal{I}+\mathcal{T})^{-1}$, which is a point that satisfies $\mathbf{v}^* = (\mathcal{I}+\mathcal{T})^{-1}(\mathbf{v}^*)$. This problem can be solved by following the recursive update $\mathbf{v}^{t+1} = (\mathcal{I}+\mathcal{T})^{-1}(\mathbf{v}^t)$. However, it can be verified that implementation of this algorithm in a distributed setting is infeasible as computing the inverse operator $(\mathcal{I}+\mathcal{T})^{-1}$ requires global communication.

To resolve this issue, we introduce a system which has the same root as $\mathcal{T}(\mathbf{v}) = \mathbf{0}$ and can be implemented in a distributed fashion. To do so, we consider the problem of finding a fixed point of the operator $(\mathbf{D}+\mathcal{T})^{-1}\mathbf{D}$ instead of $(\mathcal{I}+\mathcal{T})^{-1}$, where $\mathbf{D} \succ \mathbf{0}$ is a positive definite matrix. Note that if $\mathbf{v}^*$ is a fixed point of $(\mathbf{D}+\mathcal{T})^{-1}\mathbf{D}$, i.e., $\mathbf{v}^* = (\mathbf{D}+\mathcal{T})^{-1}\mathbf{D}(\mathbf{v}^*)$ then it satisfies the condition $\mathbf{D}\mathbf{v}^* + \mathcal{T}\mathbf{v}^* = \mathbf{D}\mathbf{v}^*$ which implies that $\mathbf{v}^*$ is a root of $\mathcal{T}$. Hence, if the operator $(\mathbf{D}+\mathcal{T})^{-1}$ is well-defined, by updating the iterates based on the following fixed point iteration

$$\mathbf{v}^{t+1} = (\mathbf{D}+\mathcal{T})^{-1}\mathbf{D}\mathbf{v}^t, \tag{10}$$

we can find a fixed point of $(\mathbf{D}+\mathcal{T})^{-1}\mathbf{D}$ and consequently a root of the operator $\mathcal{T}$. Later in Section 4, we show that the operator $(\mathbf{D}+\mathcal{T})^{-1}$ is well-defined (Lemma 1) and prove by following the update in (10), the iterates converge to a fixed point of $(\mathbf{D}+\mathcal{T})^{-1}\mathbf{D}$. We would like to highlight that the update in (10) can be interpreted as performing a proximal-point update, and for this reason we refer to our proposed method as Decentralized Proximal Point for Saddle Point problems (DPPSP). To ensure that the update in (10) can be implemented in a distributed manner we define $\mathbf{D}$ as

$$\mathbf{D} \triangleq \begin{bmatrix} \mathbf{I} & \mathbf{U} \\ \mathbf{U} & \mathbf{I} \end{bmatrix}. \tag{11}$$

To check that $\mathbf{D}$ is positive definite note that, based on Schur complement, $\mathbf{D} \succ \mathbf{0}$ holds if and only if $\mathbf{I} - \mathbf{U}^2 = \hat{\mathbf{W}} \succ \mathbf{0}$, which is satisfied based on the last condition in (7). The operator $(\mathbf{D}+\mathcal{T})^{-1}$ can be implemented in a distributed fashion, as it can be simplified to

$$(\mathbf{D}+\mathcal{T})^{-1} = \begin{bmatrix} (\alpha[\mathcal{B}+\mathcal{R}]+\mathcal{I})^{-1} & (\alpha[\mathcal{B}+\mathcal{R}]+\mathcal{I})^{-1}\mathbf{U} \\ \mathbf{0} & \mathbf{I} \end{bmatrix}. \tag{12}$$

The operator $(\alpha[\mathcal{B}+\mathcal{R}]+\mathcal{I})^{-1}$ has a block diagonal structure and can be computed locally. Moreover, the operator $\mathbf{U}$ is graph sparse and can be implemented by exchanging information among neighboring nodes. Therefore, DPPSP is a fully decentralized method. Now we proceed to formally state the local updates of nodes for implementing DPPSP. By premultiplying both sides of (10) by $(\mathbf{D}+\mathcal{T})$ and plugging in the definitions of $\mathbf{v}$, $\mathbf{D}$, and $\mathcal{T}$ we obtain

$$\mathbf{z}^{t+1} + \alpha[\mathcal{B}(\mathbf{z}^{t+1}) + \mathcal{R}(\mathbf{z}^{t+1})] = \mathbf{z}^t + \mathbf{U}\mathbf{q}^t - 2\mathbf{U}\mathbf{q}^{t+1}, \tag{13}$$

$$\mathbf{q}^{t+1} = \mathbf{U}\mathbf{z}^t + \mathbf{q}^t. \tag{14}$$

Substitute $\mathbf{q}^{t+1}$ in (13) by its update in (14) and use the definition $\mathbf{U}^2 = \mathbf{I} - \hat{\mathbf{W}}$ to obtain

$$\mathbf{z}^{t+1} + \alpha[\mathcal{B}(\mathbf{z}^{t+1}) + \mathcal{R}(\mathbf{z}^{t+1})] = (2\hat{\mathbf{W}} - \mathbf{I})\mathbf{z}^t - \mathbf{U}\mathbf{q}^t, \tag{15}$$

$$\mathbf{q}^{t+1} = \mathbf{U}\mathbf{z}^t + \mathbf{q}^t. \tag{16}$$

By subtracting two consecutive updates of $\mathbf{z}$ we can eliminate $\mathbf{q}$ from the update of $\mathbf{z}$ and obtain

$$\mathbf{z}^{t+1} + \alpha[\mathcal{B}(\mathbf{z}^{t+1}) + \mathcal{R}(\mathbf{z}^{t+1})] = 2\hat{\mathbf{W}}\mathbf{z}^t - \hat{\mathbf{W}}\mathbf{z}^{t-1} + \alpha[\mathcal{B}(\mathbf{z}^t) + \mathcal{R}(\mathbf{z}^t)], \tag{17}$$

for $t \geq 1$. By setting $\mathbf{q}^0 = \mathbf{0}$, the update for $t = 0$ is given by $\mathbf{z}^1 + \alpha[\mathcal{B}(\mathbf{z}^1) + \mathcal{R}(\mathbf{z}^1)] = (2\hat{\mathbf{W}} - \mathbf{I})\mathbf{z}^0$. Note as the mixing matrix $\hat{\mathbf{W}}$ has the sparsity pattern of the graph, computation of $2\hat{\mathbf{W}}\mathbf{z}^t$ and $\hat{\mathbf{W}}\mathbf{z}^{t-1}$ in (17) can be done in a distributed manner. Further, if the operator $(\mathcal{I} + \alpha(\mathcal{B} + \mathcal{R}))^{-1}$ is well-defined, then it can be implemented in a distributed fashion as both $\mathcal{B}$ and $\mathcal{R}$ have a block diagonal structure. The steps of the DPPSP method are summarized in Algorithm 1.

4. Theoretical Results

In this section, we first show that the inverse operators $(\mathbf{D} + \mathcal{T})^{-1}$ and $(\mathbf{I} + \alpha(\mathcal{B} + \mathcal{R}))^{-1}$ are well-defined for a properly chosen α. Then, we characterize convergence properties of DPPSP.

Lemma 1 *Consider the definitions of the operator $\mathcal{T}$ in (9) and the matrix $\mathbf{D}$ in (11). Then,*

(i) The operator $\mathcal{B} + \mathcal{R}$ is ρ-weakly monotone.

(ii) If we set $\alpha < \rho^{-1}$, then the operator $(\mathbf{I} + \alpha(\mathcal{B} + \mathcal{R}))^{-1}$ is well-defined.

(iii) If $\alpha \leq (1 - (1 - \lambda_{min}(\mathbf{W}))^{1/2})/(2\rho)$, then $\mathbf{D} + \mathcal{T}$ is $(\lambda_{min}(\mathbf{W})/4)$-strongly monotone.

The results in Lemma 1 show that if the stepsize α is properly chosen then both operators $(\mathbf{D} + \mathcal{T})^{-1}$ and $(\mathbf{I} + \alpha(\mathcal{B} + \mathcal{R}))^{-1}$ are well-defined. Hence, the updates in (10) and (17) are well-defined.

Theorem 2 *Consider the DPPSP method outlined in Algorithm 1. Suppose the conditions in Assumption 1-2 are satisfied and the stepsize is chosen such that $\alpha \leq 1/(2\rho)$. If we run DPPSP for T iterations and choose one of the iterates s uniformly at random form the time indices $1, \dots T$, then*

$$\mathbb{E}_s\left[\left\|\sum_{n=1}^{N} \mathcal{B}_n(\mathbf{z}_n^s) + \mathcal{R}_n(\mathbf{z}_n^s)\right\|\right] \leq \frac{1}{\alpha}\sqrt{\frac{N}{\lambda_{\min}(\hat{\mathbf{W}})}}\left(\frac{\|\boldsymbol{\phi}^0 - \boldsymbol{\phi}^*\|_{\mathbf{M}}}{\sqrt{T}} + \sqrt{2\alpha\rho N}D\right), \tag{18}$$

$$\mathbb{E}_s[\|\mathbf{U}\mathbf{z}^{s+1}\|] \leq \frac{\|\boldsymbol{\phi}^0 - \boldsymbol{\phi}^*\|_{\mathbf{M}}}{\sqrt{T}} + \sqrt{2\alpha\rho N}D, \tag{19}$$

where $\|\boldsymbol{\phi}^0 - \boldsymbol{\phi}^*\|_{\mathbf{M}} = \|\mathbf{z}_0 - \mathbf{z}^*\|_{\hat{\mathbf{W}}} + \|\mathbf{U}\mathbf{z}_0 - \mathbf{q}^*\|$, and $\mathbf{z}^*$ and $\mathbf{q}^*$ are defined in (8).

The result in Theorem 2 shows that after T iterations if we choose one of the iterates uniformly at random, then in expectation the iterates satisfy first-order optimality condition up to an error of $\mathcal{O}(\sqrt{\rho}+1/\sqrt{T})$, i.e., $\mathbb{E}_s[\|\sum_{n=1}^N \mathcal{B}_n(\mathbf{z}_n^s)+\mathcal{R}_n(\mathbf{z}_n^s)\|] \leq \mathcal{O}(\sqrt{\rho}+1/\sqrt{T})$, and the expected consensus error is also of $\mathcal{O}(\sqrt{\rho}+1/\sqrt{T})$, i.e., $\mathbb{E}_s[\|\mathbf{U}\mathbf{z}^{s+1}\|] \leq \mathcal{O}(\sqrt{\rho}+1/\sqrt{T})$. Exact convergence is shown in the following Theorem under the additional Assumption of Minty VI (Assumption 3)

Theorem 3 *Consider the DPPSP method outlined in Algorithm 1. Suppose Assumption 1-3 hold and the stepsize is chosen such that $\alpha = 1/(2\rho)$. If we run the DPPSP algorithm for T iterations and choose one of the iterates s uniformly at random form the time indices $1, \ldots, T$ then we have*

$$\mathbb{E}_s \left[\left\| \sum_{n=1}^{N} \mathcal{B}_n(\mathbf{z}_n^{s+1}) + \mathcal{R}_n(\mathbf{z}_n^{s+1}) \right\| \right] \leq \frac{ND}{\alpha\sqrt{T}}, \qquad \mathbb{E}_s \left[\|\mathbf{U}\mathbf{z}^s\| \right] \leq \frac{\sqrt{N}D}{\sqrt{T}}. \tag{20}$$

Theorem 3 shows that once the MVI assumption holds, the iterates generated by DPPSP can achieve any arbitrary ϵ accuracy. In particular, they find a solution with an ϵ-first-order optimality gap and an ϵ-consensus error after at most $\mathcal{O}(1/\epsilon^2)$ iterations. The convergence result in Theorem 2, which does not require the MVI assumption, depends on the graph structure (through $\lambda_{\min}(W)$). However, in Assumption 3, we assume that the point $\mathbf{z}^*$ which satisfies MVI is common to all local operators. This leads to the convergence rates in Theorem 3 that are independent of the graph structure.

References

[1] cs231n. https://cs231n.github.io/assignments2019/assignment3/.

[2] Leonard Adolphs, Hadi Daneshmand, Aurelien Lucchi, and Thomas Hofmann. Local saddle point optimization: A curvature exploitation approach. *arXiv preprint arXiv:1805.05751*, 2018.

[3] Stephen Boyd, Neal Parikh, Eric Chu, Borja Peleato, and Jonathan Eckstein. Distributed optimization and statistical learning via the alternating direction method of multipliers. *Foundations and Trends® in Machine Learning*, 3(1):1–122, 2011.

[4] Ting-Hau Chang, Mingyi Hong, and Xiongfei Wang. Multi-agent distributed optimization via inexact consensus admm. *Signal Processing, IEEE Transactions on*, 63(2):482–497, 2015.

[5] Yunmei Chen, Guanghui Lan, and Yuyuan Ouyang. Optimal primal-dual methods for a class of saddle point problems. *SIAM Journal on Optimization*, 24(4):1779–1814, 2014.

[6] Cong D Dang and Guanghui Lan. On the convergence properties of non-euclidean extragradient methods for variational inequalities with generalized monotone operators. *Computational Optimization and applications*, 60(2):277–310, 2015.

[7] Constantinos Daskalakis and Ioannis Panageas. The limit points of (optimistic) gradient descent in min-max optimization. In *Advances in Neural Information Processing Systems*, pages 9236–9246, 2018.

[8] John C Duchi, Alekh Agarwal, and Martin J Wainwright. Dual averaging for distributed optimization: convergence analysis and network scaling. *Automatic Control, IEEE Trans. on*, 57(3):592–606, 2012.

[9] Gauthier Gidel, Hugo Berard, Gaëtan Vignoud, Pascal Vincent, and Simon Lacoste-Julien. A variational inequality perspective on generative adversarial networks. *arXiv preprint arXiv:1802.10551*, 2018.

[10] Mingyi Hong, Davood Hajinezhad, and Ming-Min Zhao. Prox-pda: The proximal primal-dual algorithm for fast distributed nonconvex optimization and learning over networks. In *Proceedings of the 34th International Conference on Machine Learning - Volume 70*, ICML'17, pages 1529–1538. JMLR.org, 2017. URL http://dl.acm.org/citation.cfm?id=3305381.3305539.

[11] Mingyi Hong, Jason D Lee, and Meisam Razaviyayn. Gradient primal-dual algorithm converges to second-order stationary solutions for nonconvex distributed optimization. *arXiv preprint arXiv:1802.08941*, 2018.

[12] Dusan Jakovetic, Joao Xavier, and José MF Moura. Fast distributed gradient methods. *Automatic Control, IEEE Transactions on*, 59(5):1131–1146, 2014.

[13] Chi Jin, Praneeth Netrapalli, and Michael I. Jordan. Minmax optimization: Stable limit points of gradient descent ascent are locally optimal, 2019.

[14] Weiwei Kong and Renato DC Monteiro. An accelerated inexact proximal point method for solving nonconvex-concave min-max problems. *arXiv preprint arXiv:1905.13433*, 2019.

[15] Alec Koppel, Felicia Y Jakubiec, and Alejandro Ribeiro. A saddle point algorithm for networked online convex optimization. *IEEE Transactions on Signal Processing*, 63(19):5149–5164, 2015.

[16] GM Korpelevich. The extragradient method for finding saddle points and other problems. *Matecon*, 12:747–756, 1976.

[17] Tengyuan Liang and James Stokes. Interaction matters: A note on non-asymptotic local convergence of generative adversarial networks. *arXiv preprint arXiv:1802.06132*, 2018.

[18] Qihang Lin, Mingrui Liu, Hassan Rafique, and Tianbao Yang. Solving weakly-convex-weakly-concave saddle-point problems as weakly-monotone variational inequality. *arXiv preprint arXiv:1810.10207*, 2018.

[19] Songtao Lu, Ioannis Tsaknakis, Mingyi Hong, and Yongxin Chen. Hybrid block successive approximation for one-sided non-convex min-max problems: Algorithms and applications. *arXiv preprint arXiv:1902.08294*, 2019.

[20] Bernard Martinet. Brève communication. régularisation d'inéquations variationnelles par approximations successives. *ESAIM: Mathematical Modelling and Numerical Analysis-Modélisation Mathématique et Analyse Numérique*, 4(R3):154–158, 1970.

[21] David Mateos-Núnez and Jorge Cortés. Distributed subgradient methods for saddle-point problems. In *2015 54th IEEE Conference on Decision and Control (CDC)*, pages 5462–5467. IEEE, 2015.

[22] Aryan Mokhtari and Alejandro Ribeiro. Dsa: Decentralized double stochastic averaging gradient algorithm. *The Journal of Machine Learning Research*, 17(1):2165–2199, 2016.

[23] Aryan Mokhtari, Asuman Ozdaglar, and Sarath Pattathil. Proximal point approximations achieving a convergence rate of O(1/k) for smooth convex-concave saddle point problems: Optimistic gradient and extra-gradient methods. *arXiv preprint arXiv:1906.01115*, 2019.

[24] Aryan Mokhtari, Asuman Ozdaglar, and Sarath Pattathil. A unified analysis of extra-gradient and optimistic gradient methods for saddle point problems: Proximal point approach. *arXiv preprint arXiv:1901.08511*, 2019.

[25] Renato DC Monteiro and Benar Fux Svaiter. On the complexity of the hybrid proximal extragradient method for the iterates and the ergodic mean. *SIAM Journal on Optimization*, 20 (6):2755–2787, 2010.

[26] Vaishnavh Nagarajan and J Zico Kolter. Gradient descent gan optimization is locally stable. In *Advances in Neural Information Processing Systems*, pages 5585–5595, 2017.

[27] Angelia Nedic and Asuman Ozdaglar. Distributed subgradient methods for multi-agent optimization. *Automatic Control, IEEE Transactions on*, 54(1):48–61, 2009.

[28] Arkadi Nemirovski. Prox-method with rate of convergence o (1/t) for variational inequalities with lipschitz continuous monotone operators and smooth convex-concave saddle point problems. *SIAM Journal on Optimization*, 15(1):229–251, 2004.

[29] Arkadi Nemirovski, Anatoli Juditsky, Guanghui Lan, and Alexander Shapiro. Robust stochastic approximation approach to stochastic programming. *SIAM Journal on optimization*, 19(4): 1574–1609, 2009.

[30] Maher Nouiehed, Maziar Sanjabi, Jason D Lee, and Meisam Razaviyayn. Solving a class of non-convex min-max games using iterative first order methods. *arXiv preprint arXiv:1902.08297*, 2019.

[31] Balamurugan Palaniappan and Francis Bach. Stochastic variance reduction methods for saddle-point problems. In *Advances in Neural Information Processing Systems*, pages 1416–1424, 2016.

[32] Hassan Rafique, Mingrui Liu, Qihang Lin, and Tianbao Yang. Non-convex min-max optimization: Provable algorithms and applications in machine learning. *arXiv preprint arXiv:1810.02060*, 2018.

[33] R Tyrrell Rockafellar. Monotone operators and the proximal point algorithm. *SIAM journal on control and optimization*, 14(5):877–898, 1976.

[34] Maziar Sanjabi, Meisam Razaviyayn, and Jason D Lee. Solving non-convex non-concave min-max games under polyak-{\L}ojasiewicz condition. *arXiv preprint arXiv:1812.02878*, 2018.

[35] Kevin Scaman, Francis R. Bach, Sébastien Bubeck, Yin Tat Lee, and Laurent Massoulié. Optimal algorithms for smooth and strongly convex distributed optimization in networks. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pages 3027–3036. JMLR. org, 2017.

[36] Kevin Scaman, Francis Bach, Sébastien Bubeck, Laurent Massoulié, and Yin Tat Lee. Optimal algorithms for non-smooth distributed optimization in networks. In *Advances in Neural Information Processing Systems*, pages 2740–2749, 2018.

[37] Ioannis D Schizas, Alejandro Ribeiro, and Georgios B Giannakis. Consensus in ad hoc wsns with noisy links–part i: Distributed estimation of deterministic signals. *Signal Processing, IEEE Transactions on*, 56(1):350–364, 2008.

[38] Gesualdo Scutari and Ying Sun. Distributed nonconvex constrained optimization over time-varying digraphs. *arXiv preprint arXiv:1809.01106*, 2018.

[39] Gesualdo Scutari, Francisco Facchinei, and Lorenzo Lampariello. Parallel and distributed methods for constrained nonconvex optimization–part i: Theory. *IEEE Transactions on Signal Processing*, 65(8):1929–1944, 2017.

[40] Zebang Shen, Aryan Mokhtari, Tengfei Zhou, Peilin Zhao, and Hui Qian. Towards more efficient stochastic decentralized learning: Faster convergence and sparse communication. In *International Conference on Machine Learning*, pages 4631–4640, 2018.

[41] Wei Shi, Qing Ling, Kun Yuan, Gang Wu, and Wotao Yin. On the linear convergence of the admm in decentralized consensus optimization. *IEEE Trans. on Signal Processing*, 62(7): 1750–1761, 2014.

[42] Wei Shi, Qing Ling, Gang Wu, and Wotao Yin. Extra: An exact first-order algorithm for decentralized consensus optimization. *SIAM Journal on Optimization*, 25(2):944–966, 2015.

[43] Wei Shi, Qing Ling, Gang Wu, and Wotao Yin. A proximal gradient algorithm for decentralized composite optimization. *IEEE Transactions on Signal Processing*, 63(22):6013–6023, 2015.

[44] Haoran Sun and Mingyi Hong. Distributed non-convex first-order optimization and information processing: Lower complexity bounds and rate optimal algorithms. *arXiv preprint arXiv:1804.02729*, 2018.

[45] Kiran K Thekumparampil, Prateek Jain, Praneeth Netrapalli, and Sewoong Oh. Efficient algorithms for smooth minimax optimization. In *Advances in Neural Information Processing Systems*, pages 12659–12670, 2019.

[46] Konstantinos I Tsianos, Sean Lawlor, and Michael G Rabbat. Push-sum distributed dual averaging for convex optimization. *CDC*, pages 5453–5458, 2012.

[47] César A Uribe, Soomin Lee, Alexander Gasnikov, and Angelia Nedić. A dual approach for optimal algorithms in distributed optimization over networks. *arXiv preprint arXiv:1809.00710*, 2018.

[48] Hoi-To Wai, Zhuoran Yang, Princeton Zhaoran Wang, and Mingyi Hong. Multi-agent reinforcement learning via double averaging primal-dual optimization. In *Advances in Neural Information Processing Systems*, pages 9649–9660, 2018.

[49] Kun Yuan, Qing Ling, and Wotao Yin. On the convergence of decentralized gradient descent. *SIAM Journal on Optimization*, 26(3):1835–1854, 2016.

[50] Jinshan Zeng and Wotao Yin. On nonconvex decentralized gradient descent. *IEEE Transactions on signal processing*, 66(11):2834–2848, 2018.

SUPPLEMENTARY MATERIAL

5. Related Work

Saddle-point problems. Min-Max optimization, also known as saddle point problem, is a well studied area and several algorithms have been proposed to solve this problem. The celebrated Proximal Point method was introduced in [20] and analyzed in [33] for the case that $f(\mathbf{x}, \mathbf{y})$ is strongly convex-strongly concave or bilinear. Following this, several inexact versions of the proximal method were proposed to solve the problem including the hybrid inexact proximal method [25], which generalizes the extragradient method studied in [16, 28]. Recently optimistic gradient descent ascent (OGDA) was proposed and analyzed for solving saddle point problems [9, 17, 23, 24]. All these works consider the setting where $f(\mathbf{x}, \mathbf{y})$ is (strongly)convex-concave. Several other papers study the setting where the objective function $f(\mathbf{x}, \mathbf{y})$ is nonconvex with respect to $\mathbf{x}$ but is concave with respect to $\mathbf{y}$ [14, 19, 30, 32, 45], or study specific cases of non-convex non-concave problems [18, 34]. In a parallel line of work, several papers including [5, 29, 31] study the stochastic version of the problem where we only have access to an unbiased estimate of the gradient and not the true gradient itself. For the general non-convex non-concave setting, most of the existing works focus on showing that the gradient descent ascent (GDA) method and its infinitesimal counterpart (GDA-dynamic) converge asymptotically to the local Nash equilibrium points [2, 7, 13, 26]. However, all the mentioned works aim at finding a saddle point in a centralized setting and cannot be applied to decentralized settings.

Decentralized optimization. There are several works which study the problem of decentralized minimization when the objective function is (strongly) convex. For such settings, methods like Decentralized Gradient Descent (DGD) [12, 27, 49], Augmented Lagrangian Method (ALM) [22, 40, 42, 43], distributed implementations of the alternating direction method of multipliers (ADMM) [3, 4, 37, 41], decentralized dual averaging [8, 46], and several dual based strategies [35, 36, 47] are proposed and their corresponding convergence guarantees are established. Recently, there have been some works which look at the problem of decentralized minimization when the function is non-convex and show convergence to a first-order stationary point [10, 11, 38, 39, 44, 50]. However, all these works rely crucially on the fact that the goal is minimizing a function, and the analysis cannot be easily extended to the setting of min-max optimization which we consider in this paper.

Decentralized saddle-point problem. There are multiple works which look at the decentralized saddle point problem when the objective function is convex-concave [15, 21, 48]. However, none of these works provides any convergence guarantees for non-convex non-concave problems.

6. Definitions and Assumptions

In this section, we present definitions and assumptions that we will be using throughout the paper.

Definition 4 *Consider a function $\phi : \mathbb{R}^p \to \mathbb{R}$. The function ϕ is*
(a) convex over $\mathcal{X}$ if for any $\mathbf{x}, \hat{\mathbf{x}} \in \mathcal{X}$ we have $\phi(\hat{\mathbf{x}}) \geq \phi(\mathbf{x}) + \langle \nabla\phi(\mathbf{x}), \hat{\mathbf{x}} - \mathbf{x} \rangle$.

(b) μ-strongly convex over $\mathcal{X}$ if for any $\mathbf{x}, \hat{\mathbf{x}} \in \mathcal{X}$ we have $\phi(\hat{\mathbf{x}}) \geq \phi(\mathbf{x}) + \langle \nabla\phi(\mathbf{x}), \hat{\mathbf{x}} - \mathbf{x} \rangle + \frac{\mu}{2}\|\hat{\mathbf{x}} - \mathbf{x}\|^2$.

(c) ρ-weakly convex over $\mathcal{X}$ if for any $\mathbf{x}, \hat{\mathbf{x}} \in \mathcal{X}$ we have if $\phi(\hat{\mathbf{x}}) \geq \phi(\mathbf{x}) + \langle \nabla\phi(\mathbf{x}), \hat{\mathbf{x}} - \mathbf{x} \rangle - \frac{\rho}{2}\|\hat{\mathbf{x}} - \mathbf{x}\|^2$.

Further, a function $\phi(\mathbf{x})$ is concave, μ-strongly concave, or ρ-weakly concave, if $-\phi(\mathbf{x})$ is convex, μ-strongly convex, or ρ-weakly convex, respectively.

Definition 5 *Consider an operator $F : \mathbb{R}^d \to \mathbb{R}^d$. The operator F is*
(a) monotone over $\mathcal{Z}$ if for any $\mathbf{z}, \hat{\mathbf{z}} \in \mathcal{Z}$ we have $\langle F(\mathbf{z}) - F(\hat{\mathbf{z}}), \mathbf{z} - \hat{\mathbf{z}} \rangle \geq 0$.
(b) μ-strongly monotone over $\mathcal{Z}$ if for any $\mathbf{z}, \hat{\mathbf{z}} \in \mathcal{Z}$ we have $\langle F(\mathbf{z}) - F(\hat{\mathbf{z}}), \mathbf{z} - \hat{\mathbf{z}} \rangle \geq \mu \|\mathbf{z} - \hat{\mathbf{z}}\|^2$.
(c) ρ-weakly monotone over $\mathcal{Z}$ if for any $\mathbf{z}, \hat{\mathbf{z}} \in \mathcal{Z}$ we have [1] $\langle F(\mathbf{z}) - F(\hat{\mathbf{z}}), \mathbf{z} - \hat{\mathbf{z}} \rangle \geq -\rho \|\mathbf{z} - \hat{\mathbf{z}}\|^2$.

Definition 6 *We call the inverse operator $\mathcal{G}^{-1}$ well-defined, if for any $\mathbf{z}$ the problem of finding $\mathbf{u}$ such that $\mathbf{u} = \mathcal{G}^{-1}(\mathbf{z})$ has a unique solution. As a consequence, the operator $(\mathcal{I} + \mathcal{F})^{-1}$ is well-defined if the norm of the operator $\mathcal{F}$ is strictly smaller than 1, i.e., $\|\mathcal{F}(\mathbf{z})\| < \|\mathbf{z}\|$ for any $\mathbf{z}$.*

Now we are at the right point to state the assumptions that we will use in the rest of the paper.

Assumption 1 *The objective function $f(\mathbf{x}, \mathbf{y})$ is ρ-weakly convex with respect to $\mathbf{x}$ and ρ-weakly concave with respect to $\mathbf{y}$ for all $\mathbf{x} \in \mathcal{X}$ and $\mathbf{y} \in \mathcal{Y}$.*

Assumption 2 *The sets $\mathcal{X}$ and $\mathcal{Y}$ are convex, closed and bounded.*

Next, we characterize the properties of $F([\mathbf{x}, \mathbf{y}]) = [\nabla_{\mathbf{x}} f(\mathbf{x}, \mathbf{y}); -\nabla_{\mathbf{y}} f(\mathbf{x}, \mathbf{y})]$.

Lemma 7 *Consider the variable $\mathbf{z} = [\mathbf{x}; \mathbf{y}]$ and the operator $F(\mathbf{z}) = [\nabla_{\mathbf{x}} f(\mathbf{x}, \mathbf{y}); -\nabla_{\mathbf{y}} f(\mathbf{x}, \mathbf{y})]$, and the set $\mathcal{Z} = \mathcal{X} \times \mathcal{Y}$. If the conditions in Assumptions 1-2 are satisfied then the operator F is ρ-weakly monotone over the set $\mathcal{Z}$, and $\mathcal{Z}$ is convex, closed and bounded with a diameter D.*

Assumption 3 *(Minty Variational Inequality [MVI]): There exists a point $\mathbf{z}^*$ such that for every local operator $\mathcal{B}_n$, $\mathcal{B}_n(\mathbf{z})^\top (\mathbf{z} - \mathbf{z}^*) \geq 0$ for any $\mathbf{z} \in \mathcal{Z}$.*

This is a standard assumption which is made for non-convex optimization problems (see [6], [18] for more details). This assumption is clearly satisfied when the operator is monotone. Pseudo-monotone operators, i.e., operators which satisfy the property $\langle F(y), x - y \rangle \geq 0 \implies \langle F(x), x - y \rangle \geq 0$, also satisfy this condition. For more examples of operators that satisfy the MVI condition check [6].

7. Numerical Experiments

In this section, we show the empirical performance of DPPSP in the context of GAN training We use the number of iterations to measure the computational cost of each algorithm. The number of samples used per-iteration are identical for all settings and hence the the number of iterations is proportional to the wall-time. To assess the capability of generators in training, we run the code downloaded from [1] to produce a pair of "optimal" discriminator and generator, which are able to generate high-quality and diversified images. We use the optimal discriminator to evaluate the performance of the local generators. The global g_{loss} is defined to be the average of local losses. Note that we do not use the optimal discriminator and optimal generator to guide network training.

1. We only need access to an upper bound for the weak-monotonicity parameter ρ (as in standard convex/nonconvex optimization settings where we need an upper bound for the smoothness parameter L), and we don't need access the smallest ρ possible. Indeed, finding the best ρ would lead to the best theoretical result (as in standard optimization that knowing the best Lipschitz constant L would lead to the best theoretical guarantee). Also, we'd like to add that when the function is L-smooth, a trivial upper bound for the parameter ρ is L.

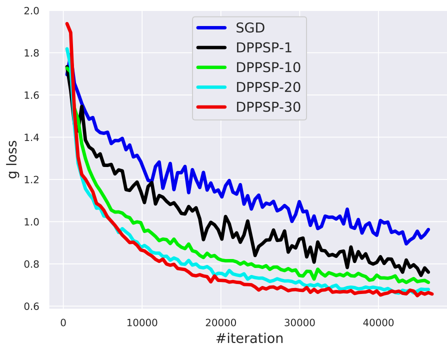


Figure 1: Generator Loss

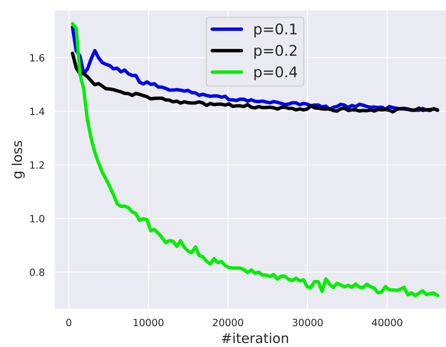


Figure 2: Different Sparsity

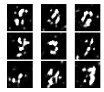
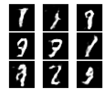
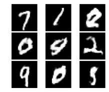
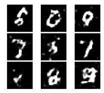
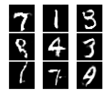
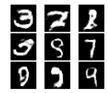
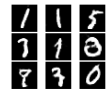
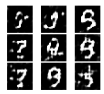
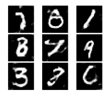
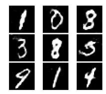
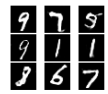
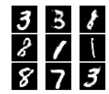
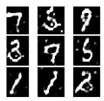
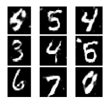
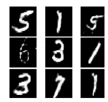
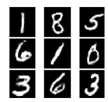
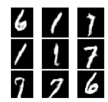
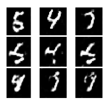
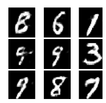
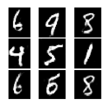
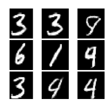
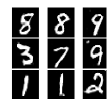
In all experiments, we assign $\rho = 25$. Our experiments are conducted on MNIST dataset as well as celebA dataset.

We use SGD as baseline to validate the performance of the proposed DPPSP method. In Figures 1 and 2, we show the performances of DPPSP with varying number of nodes and different graph connectivity on the MNIST dataset. Single-node DPPSP method outperforms SGD. The advantage of DPPSP is more significant when we have more processing nodes. Note that the number of samples used per iteration are identical for all the considered settings. Concretely, Figure 1 shows the (i) the advantage of DPPSP holds over SGD (even when $N = 1$) and (ii) how its performance improves when we have more nodes for MNIST dataset. SGD is run on a single machine which stores all of the training data. Single-node DPPSP has $N = 1$ and $\mathbf{W} = 1$. It also has access to all data like SGD. For $N > 1$, the training data is randomly and equally split to the N machines. We generate graph edges with probability 0.4 and set $\mathbf{W} = \mathbf{I} - \mathbf{L}/\tau$, where $\mathbf{L}$ is the Laplacian matrix and $\tau \geq \lambda_{max}(\mathbf{L})/2$ is a scaling parameter. The graph structure is constant throughout training. We conduct extensive search over hyperparameter α and report best results. It can be seen that the single-node DPPSP method outperforms SGD. The advantage of DPPSP gets clearer when we have more nodes, which shows the scalability of DPPSP. We can also observe that DPPSP becomes more stable when more nodes are available in the network. Note that Equation (17) is identical to a proximal problem, i.e., $\mathbf{z}_{t+1} = \text{argmin}_{\mathbf{z}} f_i(\mathbf{z}) + \frac{1}{2\alpha}\|\mathbf{z} - \mathbf{z}_t\|^2$. Now, using the assumption that f_i is weakly convex, we can construct a strongly convex problem by choosing an appropriate α. The number of steps used to solve this strongly convex problem is a hyper-parameter and empirically selected. This extra computation is considered in the runtime comparison, i.e., each 'iteration' in the comparison of DPPSP and SGD comprises of gradient steps.

To show the impact of the graph structure, we vary the edge sparsity and test the proposed DPPSP method on ten-node graphs. The edges are generated randomly with probabilities 0.1, 0.2, and 0.4 with corresponding edge numbers of 21, 27, and 45, respectively. As we observe in Figure 2, DPPSP with $p = 0.4$ has the best graph connectivity and hence has the best performance.

To better illustrate the advantage of DPPSP over SGD we compare the images produced by these two methods as time progresses. Table 1 shows images generated during training for SGD and DPPSP with 1, 10, 20, and 30 nodes. We present the images produced by these methods after 2340, 4680, 9360, 18720, and 28080 iterations. We observe that after 4680 iterations, DPPSP-20 and DPPSP-30 already have generated reasonable images, in particular, images of DPPSP-30 have better

Table 1: Fake images produced by generators that are trained using SGD and variants of DPPSP.

iterations	2340	4680	9360	18720	28080
SGD					
DPPSP-1					
DPPSP-10					
DPPSP-20					
DPPSP-30					

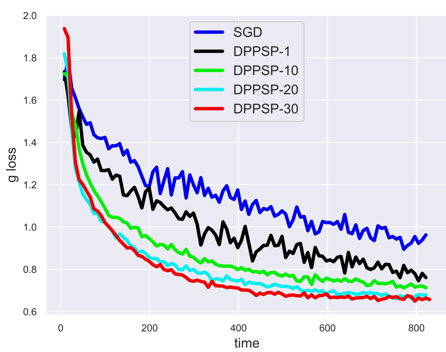


Figure 3: Generator Loss

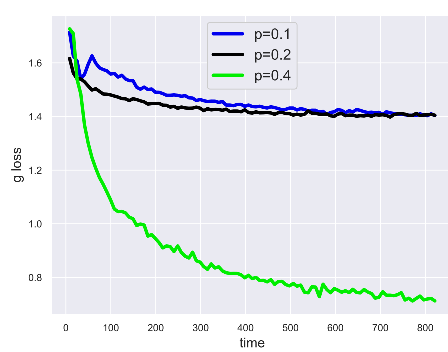


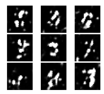
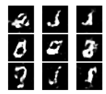
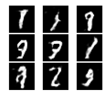
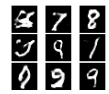
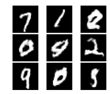
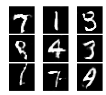
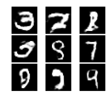
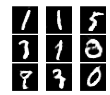
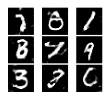
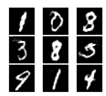
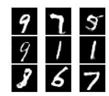
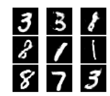
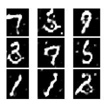
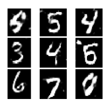
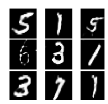
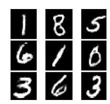
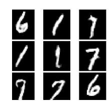
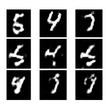
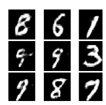
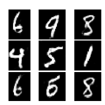
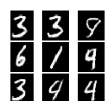
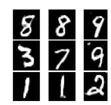
Figure 4: Different Sparsity

quality than the ones generated by DPPSP-20. Generators trained by SGD and DPPSP-1 output unsatisfactory samples even after 28080 iterations. According to these results, increasing the number of processing units not only leads to a smaller loss for the generator but also produces images with higher quality.

Instead of comparison with respect to the number of iterations, we have the comparison in terms of the actual time taken to run the experiement. This is to avoid any confusion regarding the definition of 'iteration' used in the previous plots.

We also compare SGD and DPPSP on the celebA dataset. Figure 5 demonstrates the loss of generator corresponding to SGD and DPPSP with 10 processing nodes. Similarly, here we also observe that the generator loss associated with DPPSP is smaller than the one for SGD. Also, the DPPSP algorithm is more stable compared to SGD. The minibatch size is 128 for both MNIST and celebA.

Table 2: Fake images produced by generators that are trained using SGD and variants of DPPSP.

Time	41.476s	82.953s	165.906s	331.811s	497.717s
SGD					
DPPSP-1					
DPPSP-10					
DPPSP-20					
DPPSP-30					

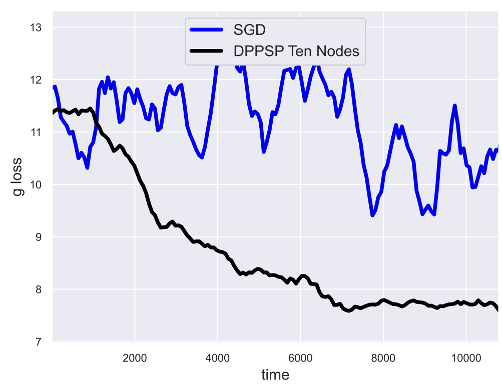


Figure 5: Performance (Generator Loss) of DPPSP with 10 nodes on the celebA dataset.

8. Proofs

8.1. Proof of Lemma 1

First, we show that the operator $\mathcal{B} + \mathcal{R}$ is ρ-weakly monotone. To do so, note that

$$
\begin{aligned}
&\langle (\mathcal{B} + \mathcal{R})(\mathbf{z}_1) - (\mathcal{B} + \mathcal{R})(\mathbf{z}_2), \mathbf{z}_1 - \mathbf{z}_2 \rangle \\
&\quad = \langle \mathcal{B}(\mathbf{z}_1) - \mathcal{B}(\mathbf{z}_2), \mathbf{z}_1 - \mathbf{z}_2 \rangle + \langle \mathcal{R}(\mathbf{z}_1) - \mathcal{R}(\mathbf{z}_2), \mathbf{z}_1 - \mathbf{z}_2 \rangle \\
&\quad \geq \langle \mathcal{B}(\mathbf{z}_1) - \mathcal{B}(\mathbf{z}_2), \mathbf{z}_1 - \mathbf{z}_2 \rangle \\
&\quad \geq -\rho \|\mathbf{z}_1 - \mathbf{z}_2\|^2
\end{aligned}
\tag{21}
$$

where the first inequality holds due to the fact that the sets $\mathcal{X}$ and $\mathcal{Y}$ are convex and therefore their indicator functions are also convex which implies that the operator $\mathcal{R}$ is monotone, and the second inequality follows from the fact that operator $\mathcal{B}$ is ρ-weakly monotno. Therefore, the operator $\mathcal{B}+\mathcal{R}$ is ρ-weakly monotone, and if we choose $\alpha < 1/\rho$ then $(\mathcal{I}+\alpha(\mathcal{B}+\mathcal{R}))^{-1}$ is well-defined.

Now we proceed to show that the operator $\mathcal{E} = \mathbf{D} + \mathcal{T}$ is strongly monotone. Note that we can write the operator $\mathcal{E}$ as

$$\mathcal{E}(\mathbf{v}) = \Bigg(\underbrace{\begin{bmatrix}\alpha[\mathcal{B}+\mathcal{R}] & 0\\ 0 & 0\end{bmatrix}}_{\mathcal{E}_1} + \underbrace{\begin{bmatrix}\mathbf{I} & 2\mathbf{U}\\ \mathbf{0} & \mathbf{I}\end{bmatrix}}_{\mathcal{E}_2}\Bigg)\underbrace{\begin{bmatrix}\mathbf{z}\\ \mathbf{q}\end{bmatrix}}_{\mathbf{v}}, \tag{22}$$

Based on the first part of the proof we know that $\alpha(\mathcal{B}+\mathcal{R})$ is $\alpha\rho$-weakly monotone. Therefore, the operator $\mathcal{E}_1$ is also $\alpha\rho$-weakly monotone. Now we proceed to show that $\mathcal{E}_2$ is strongly monotone. We can write

$$\begin{aligned}
&\langle \mathcal{E}_2(\mathbf{v}_2) - \mathcal{E}_2(\mathbf{v}_2), \mathbf{v}_2 - \mathbf{v}_1\rangle\\
&\quad= \langle (\mathbf{z}_2 + 2\mathbf{U}\mathbf{q}_2) - (\mathbf{z}_1 + 2\mathbf{U}\mathbf{q}_1), \mathbf{z}_2 - \mathbf{z}_1\rangle + \langle \mathbf{q}_2 - \mathbf{q}_1, \mathbf{q}_2 - \mathbf{q}_1\rangle\\
&\quad= \langle \mathbf{z}_2 - \mathbf{z}_1, \mathbf{z}_2 - \mathbf{z}_1\rangle + 2\langle \mathbf{U}(\mathbf{q}_2 - \mathbf{q}_1), \mathbf{z}_2 - \mathbf{z}_1\rangle + \langle \mathbf{q}_2 - \mathbf{q}_1, \mathbf{q}_2 - \mathbf{q}_1\rangle\\
&\quad\geq \|\mathbf{z}_2 - \mathbf{z}_1\|^2 - 2\|\mathbf{U}\|\|\mathbf{q}_2 - \mathbf{q}_1\|\|\mathbf{z}_2 - \mathbf{z}_1\| + \|\mathbf{q}_2 - \mathbf{q}_1\|^2
\end{aligned} \tag{23}$$

If we define $\beta := \max\{|\lambda_{min}(\mathbf{U})|, |\lambda_{max}(\mathbf{U})|\}$, then it can verified that $\|\mathbf{U}\| \leq \beta$. Using this inequality we can show that

$$\begin{aligned}
\langle \mathcal{E}_2(\mathbf{v}_2) - \mathcal{E}_2(\mathbf{v}_2), \mathbf{v}_2 - \mathbf{v}_1\rangle &\geq \|\mathbf{z}_2 - \mathbf{z}_1\|^2 - 2\beta\|\mathbf{q}_2 - \mathbf{q}_1\|\|\mathbf{z}_2 - \mathbf{z}_1\| + \|\mathbf{q}_2 - \mathbf{q}_1\|^2\\
&\geq \|\mathbf{z}_2 - \mathbf{z}_1\|^2 - \beta\|\mathbf{q}_2 - \mathbf{q}_1\|^2 - \beta\|\mathbf{z}_2 - \mathbf{z}_1\|^2 + \|\mathbf{q}_2 - \mathbf{q}_1\|^2\\
&= (1-\beta)\|\mathbf{z}_2 - \mathbf{z}_1\|^2 + (1-\beta)\|\mathbf{q}_2 - \mathbf{q}_1\|^2\\
&= (1-\beta)\|\mathbf{v}_2 - \mathbf{v}_1\|^2.
\end{aligned} \tag{24}$$

Now note that $\mathbf{U} = (\mathbf{I} - \hat{\mathbf{W}})^{1/2}$. Since the eigenvalues of $\mathbf{W}$ belong to the interval $(0, 1]$ it can be easily verified that $\mathbf{U}$ is positive semidefinite. We can further show that $\lambda_{max}(\mathbf{U}) = (1 - \lambda_{min}(\mathbf{W}))^{1/2}$ and $\lambda_{min}(\mathbf{U}) = (1 - \lambda_{max}(\mathbf{W}))^{1/2}$. Therefore, $\beta = (1 - \lambda_{min}(\mathbf{W}))^{1/2}$. This result implies that $\mathcal{E}_2$ is $1 - (1 - \lambda_{min}(\mathbf{W}))^{1/2}$-strongly monotone. Therefore we can write

$$\langle \mathcal{E}(\mathbf{v}_2) - \mathcal{E}(\mathbf{v}_2), \mathbf{v}_2 - \mathbf{v}_1\rangle \geq \Big(1 - (1 - \lambda_{min}(\mathbf{W}))^{1/2} - \alpha\rho\Big)\, \|\mathbf{v}_2 - \mathbf{v}_1\|^2. \tag{25}$$

Hence, if we choose the stepsize α such that

$$\alpha \; < \; \frac{1 - (1 - \lambda_{min}(\mathbf{W}))^{1/2}}{\rho}, \tag{26}$$

then $\mathcal{E}$ is strongly monotone. In particular, if we set $\alpha \leq (1 - (1 - \lambda_{min}(\mathbf{W}))^{1/2})/(2\rho)$, then $\mathcal{E}$ is $(1 - (1 - \lambda_{min}(\mathbf{W}))^{1/2})/(2)$-strongly monotone. As $1 - (1 - x)^{1/2}$ is lower bounded by $x/2$ for $x \in [0, 1]$, this result shows that the operator $\mathcal{E} = \mathbf{D} + \mathcal{T}$ is $(\lambda_{min}(\mathbf{W})/4)$-strongly monotone when $\alpha \leq (1 - (1 - \lambda_{min}(\mathbf{W}))^{1/2})/(2\rho)$.

8.2. Proof of Theorem 2

Consider the operator $\mathcal{F} \stackrel{\text{def}}{=} \mathcal{B} + \mathcal{R}$. Note that using the update rules in (15) and (16), we can show that

$$\begin{aligned}\alpha\mathcal{F}(\mathbf{z}^{t+1}) &= (2\hat{\mathbf{W}} - \mathbf{I})\mathbf{z}^t - \mathbf{z}^{t+1} - \mathbf{U}\mathbf{q}^t \\ &= (2\hat{\mathbf{W}} - \mathbf{I})\mathbf{z}^t - \mathbf{z}^{t+1} - \mathbf{U}\mathbf{q}^{t+1} + (\mathbf{I} - \hat{\mathbf{W}})\mathbf{z}^t \\ &= \hat{\mathbf{W}}\mathbf{z}^t - \mathbf{z}^{t+1} - \mathbf{U}\mathbf{q}^{t+1} \\ &= \hat{\mathbf{W}}\mathbf{z}^t - \mathbf{z}^{t+1} - \mathbf{U}\mathbf{q}^{t+2} + (\mathbf{I} - \hat{\mathbf{W}})\mathbf{z}^{t+1} \\ &= \hat{\mathbf{W}}(\mathbf{z}^t - \mathbf{z}^{t+1}) - \mathbf{U}\mathbf{q}^{t+2}. \end{aligned} \tag{27}$$

Based on the first optimality condition in (8) we know that $\alpha\mathcal{F}(\mathbf{z}^*) + \mathbf{U}\mathbf{q}^* = \mathbf{0}$. By subtracting the optimality condition from (27), we obtain that

$$\alpha[\mathcal{F}(\mathbf{z}^{t+1}) - \mathcal{F}(\mathbf{z}^*)] = \hat{\mathbf{W}}(\mathbf{z}^t - \mathbf{z}^{t+1}) - \mathbf{U}(\mathbf{q}^{t+2} - \mathbf{q}^*). \tag{28}$$

Using the result in (28), we can write

$$\begin{aligned}&\langle \mathbf{z}^{t+1} - \mathbf{z}^*, \alpha[\mathcal{F}(\mathbf{z}^*) - \mathcal{F}(\mathbf{z}^{t+1})]\rangle \\ &\quad= \langle \mathbf{z}^{t+1} - \mathbf{z}^*, -\hat{\mathbf{W}}(\mathbf{z}^t - \mathbf{z}^{t+1}) + \mathbf{U}(\mathbf{q}^{t+2} - \mathbf{q}^*)\rangle \\ &\quad= \langle \mathbf{z}^{t+1} - \mathbf{z}^*, \hat{\mathbf{W}}(\mathbf{z}^{t+1} - \mathbf{z}^t)\rangle + \langle \mathbf{z}^{t+1} - \mathbf{z}^*, \mathbf{U}(\mathbf{q}^{t+2} - \mathbf{q}^*)\rangle \\ &\quad= \langle \mathbf{z}^{t+1} - \mathbf{z}^*, \hat{\mathbf{W}}(\mathbf{z}^{t+1} - \mathbf{z}^t)\rangle + \langle \mathbf{q}^{t+2} - \mathbf{q}^{t+1}, \mathbf{q}^{t+2} - \mathbf{q}^*\rangle, \end{aligned} \tag{29}$$

where the last equality uses the definition of $\mathbf{q}^t$ and that $\mathbf{U}\mathbf{z}^* = \mathbf{0}$. By applying the generalized Law of cosines $2\langle a, b\rangle = \|a\|^2 + \|b\|^2 - \|a - b\|^2$, we can write the first inner product as

$$\langle \mathbf{z}^{t+1} - \mathbf{z}^*, \hat{\mathbf{W}}(\mathbf{z}^{t+1} - \mathbf{z}^t)\rangle = \frac{1}{2}\left(\|\mathbf{z}^{t+1} - \mathbf{z}^*\|^2_{\hat{\mathbf{W}}} + \|\mathbf{z}^{t+1} - \mathbf{z}^t\|^2_{\hat{\mathbf{W}}} - \|\mathbf{z}^t - \mathbf{z}^*\|^2_{\hat{\mathbf{W}}}\right), \tag{30}$$

and the second inner product as

$$\langle \mathbf{q}^{t+2} - \mathbf{q}^{t+1}, \mathbf{q}^{t+2} - \mathbf{q}^*\rangle = \frac{1}{2}\left(\|\mathbf{q}^{t+2} - \mathbf{q}^*\|^2 + \|\mathbf{q}^{t+2} - \mathbf{q}^{t+1}\|^2 - \|\mathbf{q}^{t+1} - \mathbf{q}^*\|^2\right), \tag{31}$$

Substitute the expressions in (30) and (31) into (29) to obtain

$$\begin{aligned}&\langle \mathbf{z}^{t+1} - \mathbf{z}^*, \alpha[\mathcal{F}(\mathbf{z}^*) - \mathcal{F}(\mathbf{z}^{t+1})]\rangle \\ &\quad= \frac{1}{2}\left(\|\mathbf{z}^{t+1} - \mathbf{z}^*\|^2_{\hat{\mathbf{W}}} + \|\mathbf{z}^{t+1} - \mathbf{z}^t\|^2_{\hat{\mathbf{W}}} - \|\mathbf{z}^t - \mathbf{z}^*\|^2_{\hat{\mathbf{W}}}\right) \\ &\qquad+ \frac{1}{2}\left(\|\mathbf{q}^{t+2} - \mathbf{q}^*\|^2 + \|\mathbf{q}^{t+2} - \mathbf{q}^{t+1}\|^2 - \|\mathbf{q}^{t+1} - \mathbf{q}^*\|^2\right) \end{aligned} \tag{32}$$

Now we define matrix $\mathbf{M} \in \mathbb{R}^{2Nd \times 2Nd}$ and sequence of vectors $\boldsymbol{\phi}^t \in \mathbb{R}^{2Nd}$ as

$$\mathbf{M} \triangleq \begin{bmatrix} \hat{\mathbf{W}} & \mathbf{0} \\ \mathbf{0} & \mathbf{I} \end{bmatrix}, \quad \boldsymbol{\phi}^t \triangleq \begin{bmatrix} \mathbf{z}^t \\ \mathbf{q}^{t+1} \end{bmatrix}. \tag{33}$$

Considering these definitions we can write the inequality in (32) as

$$\|\boldsymbol{\phi}^{t+1}-\boldsymbol{\phi}^*\|_{\mathbf{M}}^2+\|\boldsymbol{\phi}^{t+1}-\boldsymbol{\phi}^t\|_{\mathbf{M}}^2-\|\boldsymbol{\phi}^t-\boldsymbol{\phi}^*\|_{\mathbf{M}}^2=2\alpha\langle\mathbf{z}^{t+1}-\mathbf{z}^*,\mathcal{F}(\mathbf{z}^*)-\mathcal{F}(\mathbf{z}^{t+1})\rangle. \quad (34)$$

Using the fact that $\mathcal{F}=\mathcal{B}+\mathcal{R}$ is ρ-weakly monotone, we have

$$2\alpha\langle\mathbf{z}^{t+1}-\mathbf{z}^*,\mathcal{F}(\mathbf{z}^*)-\mathcal{F}(\mathbf{z}^{t+1})\rangle\leq 2\alpha\rho\|\mathbf{z}^{t+1}-\mathbf{z}^*\|^2. \quad (35)$$

Replace the upper bound in (35) into (34) and regroup the terms to obtain

$$\|\boldsymbol{\phi}^t-\boldsymbol{\phi}^*\|_{\mathbf{M}}^2-\|\boldsymbol{\phi}^{t+1}-\boldsymbol{\phi}^*\|_{\mathbf{M}}^2\geq\|\boldsymbol{\phi}^{t+1}-\boldsymbol{\phi}^t\|_{\mathbf{M}}^2-2\alpha\rho\|\mathbf{z}^{t+1}-\mathbf{z}^*\|^2. \quad (36)$$

Replace $\|\mathbf{z}^{t+1}-\mathbf{z}^*\|^2=\sum_{n=1}^N\|\mathbf{z}_n^{t+1}-\mathbf{z}_n^*\|^2$ by its upper bound ND^2 where D is the diameter of the set $\mathcal{Z}$.

$$\|\boldsymbol{\phi}^t-\boldsymbol{\phi}^*\|_{\mathbf{M}}^2-\|\boldsymbol{\phi}^{t+1}-\boldsymbol{\phi}^*\|_{\mathbf{M}}^2\geq\|\boldsymbol{\phi}^{t+1}-\boldsymbol{\phi}^t\|_{\mathbf{M}}^2-2\alpha\rho ND^2. \quad (37)$$

Sum the above inequality from $t=0$ to $T-1$ to obtain

$$\|\boldsymbol{\phi}^0-\boldsymbol{\phi}^*\|_{\mathbf{M}}^2-\|\boldsymbol{\phi}^T-\boldsymbol{\phi}^*\|_{\mathbf{M}}^2\geq\sum_{t=0}^{T-1}\|\boldsymbol{\phi}^{t+1}-\boldsymbol{\phi}^t\|_{\mathbf{M}}^2-2\alpha\rho TND^2 \quad (38)$$

which implies that

$$\frac{1}{T}\sum_{t=0}^{T-1}\|\boldsymbol{\phi}^{t+1}-\boldsymbol{\phi}^t\|_{\mathbf{M}}^2\leq\frac{\|\boldsymbol{\phi}^0-\boldsymbol{\phi}^*\|_{\mathbf{M}}^2}{T}+2\alpha\rho ND^2 \quad (39)$$

Therefore,

$$\frac{1}{T}\sum_{t=0}^{T-1}\|\mathbf{z}^{t+1}-\mathbf{z}^t\|^2\leq\frac{1}{\lambda_{\min}(\hat{\mathbf{W}})}\left(\frac{\|\boldsymbol{\phi}^0-\boldsymbol{\phi}^*\|_{\mathbf{M}}^2}{T}+2\alpha\rho ND^2\right) \quad (40)$$

and

$$\frac{1}{T}\sum_{t=0}^{T-1}\|\mathbf{q}^{t+2}-\mathbf{q}^{t+1}\|^2=\frac{1}{T}\sum_{t=0}^{T-1}\|\mathbf{U}\mathbf{z}^{t+1}\|^2\leq\frac{\|\boldsymbol{\phi}^0-\boldsymbol{\phi}^*\|_{\mathbf{M}}^2}{T}+2\alpha\rho ND^2 \quad (41)$$

Assume that we choose one of the iterates s uniformly at random from $0,\ldots,T-1$, then

$$\mathbb{E}_s[\|\mathbf{z}^{s+1}-\mathbf{z}^s\|^2]\leq\frac{1}{\lambda_{\min}(\hat{\mathbf{W}})}\left(\frac{\|\boldsymbol{\phi}^0-\boldsymbol{\phi}^*\|_{\mathbf{M}}^2}{T}+2\alpha\rho ND^2\right) \quad (42)$$

and

$$\mathbb{E}_s[\|\mathbf{U}\mathbf{z}^{s+1}\|^2]\leq\frac{\|\boldsymbol{\phi}^0-\boldsymbol{\phi}^*\|_{\mathbf{M}}^2}{T}+2\alpha\rho ND^2 \quad (43)$$

Now, using the fact that $\mathbb{E}[X]\leq\sqrt{\mathbb{E}[X^2]}$ and $\sqrt{a+b}\leq\sqrt{a}+\sqrt{b}$ for positive a,b, we have

$$\mathbb{E}_s[\|\mathbf{z}^{s+1}-\mathbf{z}^s\|]\leq\sqrt{\frac{1}{\lambda_{\min}(\hat{\mathbf{W}})}}\left(\frac{\|\boldsymbol{\phi}^0-\boldsymbol{\phi}^*\|_{\mathbf{M}}}{\sqrt{T}}+\sqrt{2\alpha\rho N}D\right) \quad (44)$$

and

$$\mathbb{E}_s[\|\mathbf{U}\mathbf{z}^{s+1}\|] \leq \frac{\|\boldsymbol{\phi}^0 - \boldsymbol{\phi}^*\|_{\mathbf{M}}}{\sqrt{T}} + \sqrt{2\alpha\rho N}D \tag{45}$$

Now based on the expression in (27) we have

$$\alpha\mathcal{F}(\mathbf{z}^{t+1}) = \hat{\mathbf{W}}(\mathbf{z}^t - \mathbf{z}^{t+1}) - \mathbf{U}\mathbf{q}^{t+2}. \tag{46}$$

Multiply both sides by the the vector $\mathbf{1}_N \otimes \mathbf{I}_d$ we obtain

$$\alpha(\mathbf{1}_N \otimes \mathbf{I}_d)^\top \mathcal{F}(\mathbf{z}^{t+1}) = (\mathbf{1}_N \otimes \mathbf{I}_d)^\top (\mathbf{z}^t - \mathbf{z}^{t+1}). \tag{47}$$

which implies that

$$\alpha \sum_{n=1}^{N} (\mathcal{B}_n(\mathbf{z}_n^{t+1}) + \mathcal{R}_n(\mathbf{z}_n^{t+1})) = \sum_{n=1}^{N} (\mathbf{z}_n^t - \mathbf{z}_n^{t+1}). \tag{48}$$

Therefore,

$$\begin{aligned}\alpha \left\| \sum_{n=1}^{N} \mathcal{B}_n(\mathbf{z}_n^{t+1}) + \mathcal{R}_n(\mathbf{z}_n^{t+1}) \right\| &\leq \sum_{n=1}^{N} \|\mathbf{z}_n^t - \mathbf{z}_n^{t+1}\| \\ &\leq \sqrt{N \sum_{n=1}^{N} \|\mathbf{z}_n^t - \mathbf{z}_n^{t+1}\|_2^2} \\ &\leq \sqrt{N}\|\mathbf{z}^t - \mathbf{z}^{t+1}\|_2 \end{aligned} \tag{49}$$

Hence, we can show that

$$\mathbb{E}_s\left[\left\| \sum_{n=1}^{N} \mathcal{B}_n(\mathbf{z}_n^{s+1}) + \mathcal{R}_n(\mathbf{z}_n^{s+1}) \right\|\right] \leq \frac{1}{\alpha}\sqrt{\frac{N}{\lambda_{\min}(\hat{\mathbf{W}})}} \left(\frac{\|\boldsymbol{\phi}^0 - \boldsymbol{\phi}^*\|_{\mathbf{M}}}{\sqrt{T}} + \sqrt{2\alpha\rho N}D \right) \tag{50}$$

8.3. Proof of Theorem 3

Consider $\hat{\mathbf{z}} = [\mathbf{z}^*; \ldots; \mathbf{z}^*]$ where $\mathbf{z}^*$ is the point that satisfies the condition in Assumption 3. Then we can show that

$$\mathcal{B}(\mathbf{z})^\top (\mathbf{z} - \hat{\mathbf{z}}^*) \geq 0, \tag{51}$$

for any $\mathbf{z} = [\mathbf{z}_1; \ldots; \mathbf{z}_N] \in \mathcal{Z}^N$. Further, note that it can be verified that $\mathcal{R}(\mathbf{z})^\top(\mathbf{z} - \hat{\mathbf{z}}^*) \geq 0$, as there always exists a subgradient which has a positive inner product with the vector $\mathbf{z} - \hat{\mathbf{z}}^*$. Hence,

$$(\mathcal{B}(\mathbf{z}) + \mathcal{R}(\mathbf{z}))^\top (\mathbf{z} - \hat{\mathbf{z}}^*) \geq 0, \tag{52}$$

for any $\mathbf{z} \in \mathcal{Z}^N$.

Now we proceed to show that there exists a vector $\hat{\mathbf{q}}^*$ such that the vector $\hat{\mathbf{v}}^* = [\hat{\mathbf{z}}^*; \hat{\mathbf{q}}^*]$ satisfies the condition

$$\mathcal{T}(\mathbf{v})^\top (\mathbf{v} - \hat{\mathbf{v}}^*) \geq 0, \tag{53}$$

for all $\mathbf{v} = [\mathbf{z}; \mathbf{q}]$ such that $\mathbf{z} \in \mathcal{Z}^N$ and $\mathbf{q} \in \mathbb{R}^{Nd}$. To do so, consider $\hat{\mathbf{q}}^*$ as a vector that belongs to the null space $\mathbf{U}$, i.e., $\mathbf{U}\hat{\mathbf{q}}^* = \mathbf{0}$. Hence, for any $\mathbf{z} \in \mathcal{Z}^N$ we have $\mathbf{z}^\top \mathbf{U}\hat{\mathbf{q}}^* = 0$. Therefore, we can write

$$\alpha(\mathcal{B}(\mathbf{z}) + \mathcal{R}(\mathbf{z}))^\top (\mathbf{z} - \hat{\mathbf{z}}^*) + \mathbf{z}^\top \mathbf{U}\hat{\mathbf{q}}^* \geq 0, \tag{54}$$

Furter, we know that $\hat{\mathbf{z}}^*$ belongs to the null space of $\mathbf{U}$ and therefore $(\hat{\mathbf{z}}^*)^\top \mathbf{U}\mathbf{q} = 0$ for any $\mathbf{q} \in \mathbb{R}^{Nd}$. Therefore, we have

$$\alpha(\mathcal{B}(\mathbf{z}) + \mathcal{R}(\mathbf{z}))^\top (\mathbf{z} - \hat{\mathbf{z}}^*) + \mathbf{z}^\top \mathbf{U}\hat{\mathbf{q}}^* - \mathbf{q}^\top \mathbf{U}\hat{\mathbf{z}}^* \geq 0, \tag{55}$$

Now add and subtract $\mathbf{z}^\top \mathbf{U}\mathbf{q}$ to the left hand side to obtain

$$\alpha(\mathcal{B}(\mathbf{z}) + \mathcal{R}(\mathbf{z}))^\top (\mathbf{z} - \hat{\mathbf{z}}^*) - \mathbf{z}^\top \mathbf{U}(\mathbf{q} - \hat{\mathbf{q}}^*) + \mathbf{q}^\top \mathbf{U}(\mathbf{z} - \hat{\mathbf{z}}^*) \geq 0, \tag{56}$$

for any $\mathbf{z} \in \mathcal{Z}^N$ and $\mathbf{q} \in \mathbb{R}^{Nd}$. This expression can also be written as

$$\begin{bmatrix} \alpha[\mathcal{B} + \mathcal{R}](\mathbf{z}) + \mathbf{U}\mathbf{q} \\ -\mathbf{U}\mathbf{z} \end{bmatrix}^\top \begin{bmatrix} \mathbf{z} - \hat{\mathbf{z}}^* \\ \mathbf{q} - \hat{\mathbf{q}}^* \end{bmatrix} \geq 0 \tag{57}$$

which is equivalent to

$$\left(\begin{bmatrix} \alpha[\mathcal{B} + \mathcal{R}] & \mathbf{U} \\ -\mathbf{U} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{z} \\ \mathbf{q} \end{bmatrix} \right)^\top \begin{bmatrix} \mathbf{z} - \hat{\mathbf{z}}^* \\ \mathbf{q} - \hat{\mathbf{q}}^* \end{bmatrix} \geq 0 \tag{58}$$

By considering the definitions of $\mathcal{T}$, $\mathbf{v}$, and $\hat{\mathbf{v}}^*$, it can be verified that (58) implies (53).

Note that the sequence of iterates generated by our proposed method satisfy the condition

$$\mathcal{T}(\mathbf{v}^{t+1}) + \mathbf{D}(\mathbf{v}^{t+1} - \mathbf{v}^t) = \mathbf{0}, \tag{59}$$

Consider the definition $\|\mathbf{a}\|_{\mathbf{A}}^2 := \mathbf{a}^\top \mathbf{A}\mathbf{a}$. Then, it can be verified that

$$\begin{aligned} \|\mathbf{v}^t - \hat{\mathbf{v}}^*\|_{\mathbf{D}}^2 &= \|\mathbf{v}^{t+1} - \hat{\mathbf{v}}^*\|_{\mathbf{D}}^2 + \|\mathbf{v}^{t+1} - \mathbf{v}^t\|_{\mathbf{D}}^2 + 2(\mathbf{v}^t - \mathbf{v}^{t+1})^T \mathbf{D}(\mathbf{v}^{t+1} - \hat{\mathbf{v}}^*) \\ &= \|\mathbf{v}^{t+1} - \hat{\mathbf{v}}^*\|_{\mathbf{D}}^2 + \|\mathbf{v}^{t+1} - \mathbf{v}^t\|_{\mathbf{D}}^2 + 2\mathcal{T}(\mathbf{v}^{t+1})^T (\mathbf{v}^{t+1} - \hat{\mathbf{v}}^*) \\ &\geq \|\mathbf{v}^{t+1} - \hat{\mathbf{v}}^*\|_{\mathbf{D}}^2 + \|\mathbf{v}^{t+1} - \mathbf{v}^t\|_{\mathbf{D}}^2 \end{aligned} \tag{60}$$

If we sum both sides from $k = 0$ to $k = T - 1$ we obtain that

$$\sum_{t=0}^{T-1} \|\mathbf{v}^{t+1} - \mathbf{v}^t\|^2 \leq \|\mathbf{v}^0 - \hat{\mathbf{v}}^*\|^2 - \|\mathbf{v}_T - \hat{\mathbf{v}}^*\|^2 \tag{61}$$

Therefore, if we choose one of the indices k from 0 to $T-1$ with probability $1/T$ the expected value of the random variable $\|\mathbf{v}^{t+1} - \mathbf{v}^t\|^2$ will be

$$\begin{aligned} \mathbb{E}_s[\|\mathbf{v}^{s+1} - \mathbf{v}^s\|^2] &= \frac{1}{T} \sum_{k=0}^{T-1} \|\mathbf{v}^{t+1} - \mathbf{v}^t\|^2 \\ &\leq \frac{\|\mathbf{v}^0 - \hat{\mathbf{v}}^*\|^2}{T} \end{aligned} \tag{62}$$

Based on the equality in (59) we can show that

$$\begin{bmatrix} \alpha[\mathcal{B}+\mathcal{R}](\mathbf{z}^{t+1}) + \mathbf{U}\mathbf{q}^{t+1} \\ -\mathbf{U}\mathbf{z}^{t+1} \end{bmatrix} = \begin{bmatrix} \mathbf{z}^t - \mathbf{z}^{t+1} + \mathbf{U}(\mathbf{q}^t - \mathbf{q}^{t+1}) \\ \mathbf{U}(\mathbf{z}^t - \mathbf{z}^{t+1}) + \mathbf{q}^t - \mathbf{q}^{t+1} \end{bmatrix} \tag{63}$$

Multiply both sides of the first equality by the vector $\mathbf{1}_N \otimes \mathbf{I}_d$ to obtain which implies that

$$\alpha \sum_{n=1}^{N} (\mathcal{B}_n(\mathbf{z}_n^{t+1}) + \mathcal{R}_n(\mathbf{z}_n^{t+1})) = \sum_{n=1}^{N} (\mathbf{z}_n^t - \mathbf{z}_n^{t+1}). \tag{64}$$

Therefore,

$$\begin{aligned} \alpha \left\| \sum_{n=1}^{N} \mathcal{B}_n(\mathbf{z}_n^{t+1}) + \mathcal{R}_n(\mathbf{z}_n^{t+1}) \right\| &\leq \sum_{n=1}^{N} \|\mathbf{z}_n^t - \mathbf{z}_n^{t+1}\| \\ &\leq \sqrt{N \sum_{n=1}^{N} \|\mathbf{z}_n^t - \mathbf{z}_n^{t+1}\|_2^2} \\ &\leq \sqrt{N} \|\mathbf{z}^t - \mathbf{z}^{t+1}\|_2 \end{aligned} \tag{65}$$

Hence, we can show that

$$\mathbb{E}_s \left[\left\| \sum_{n=1}^{N} \mathcal{B}_n(\mathbf{z}_n^{s+1}) + \mathcal{R}_n(\mathbf{z}_n^{s+1}) \right\|^2 \right] \leq \frac{N}{\alpha^2} \frac{\|\mathbf{v}^0 - \hat{\mathbf{v}}^*\|^2}{T} \tag{66}$$

Further, we can show that

$$\begin{aligned} \mathbb{E}_s[\|\mathbf{U}\mathbf{z}^s\|^2] &= \frac{1}{T} \sum_{t=0}^{T-1} \|\mathbf{U}\mathbf{z}^s\|^2 \\ &= \frac{1}{T} \sum_{t=0}^{T-1} \|\mathbf{q}^{t+1} - \mathbf{q}^t\|^2 \\ &\leq \frac{\|\mathbf{v}^0 - \hat{\mathbf{v}}^*\|^2}{T} \end{aligned} \tag{67}$$

As we simply can choose $\hat{\mathbf{q}}^*$ as $\hat{\mathbf{q}}^* = 0$ and the initial iterate is $\mathbf{q}^0 = 0$ we can simplify $\|\mathbf{v}^0 - \hat{\mathbf{v}}^*\|^2$ as $\|\mathbf{z}^0 - \hat{\mathbf{z}}^*\|^2$ in both (66) and (67).