A Unified and Refined Convergence Analysis for Non-Convex Decentralized Learning

Sulaiman A. Alghunaim $^*$ and Kun Yuan $^*$, Member, IEEE

Abstract—We study the consensus decentralized optimization problem where the objective function is the average of $n$ agents private non-convex cost functions; moreover, the agents can only communicate to their neighbors on a given network topology. The stochastic learning setting is considered in this paper where each agent can only access a noisy estimate of its gradient. Many decentralized methods can solve such problem including EXTRA, Exact-Diffusion/D$^2$, and gradient-tracking. Unlike the famed D$^2$GD algorithm, these methods have been shown to be robust to the heterogeneity across the local cost functions. However, the established convergence rates for these methods indicate that their sensitivity to the network topology is worse than D$^2$GD. Such theoretical results imply that these methods can perform much worse than D$^2$GD on sparse networks, which, however, contradicts empirical experiments where D$^2$GD is observed to be more sensitive to the network topology. In this work, we study a general stochastic unified decentralized algorithm (SUDA) that includes the above methods as special cases. We establish the convergence of SUDA under both non-convex and the Polyak-Łojasiewicz condition settings. Our results provide improved network topology dependent bounds for these methods (such as Exact-Diffusion/D$^2$ and gradient-tracking) compared with existing literature. Moreover, our results show that these methods are often less sensitive to the network topology compared to D$^2$GD, which agrees with numerical experiments.

Index Terms—Adaptation and learning, decentralized optimization, distributed processing, exact diffusion, EXTRA, gradient tracking.

I. INTRODUCTION

In a distributed multi-agent optimization problem, the inputs (e.g., functions, variables, data) are spread over multiple computing agents (e.g., nodes, processors) that are connected over some network, and the agents are required to communicate with each other to solve this problem. Distributed optimization has attracted a lot of attention due to the need of developing efficient methods to solve large-scale optimization problems [1] such as in deep neural networks applications [2]. Decentralized optimization methods are algorithms where the agents seek to find a solution through local interactions (dictated by the network connection) with their neighboring agents. Decentralized methods have several advantages over centralized methods, which require all the agents to communicate with a central coordinator, such as their robustness to failure and privacy. Moreover, decentralized methods have been shown to enjoy lower communication cost compared to centralized methods under certain practical scenarios [3]–[5].

In this work, we consider a network (graph) of $n$ collaborative agents (nodes) that are interested in solving the following distributed stochastic optimization problem:

$$\min_{x \in \mathbb{R}^d} f(x) = \frac{1}{n} \sum_{i=1}^{n} f_i(x), \quad f_i(x) \triangleq \mathbb{E}_{\xi_i} [F_i(x; \xi_i)].$$

In the above formulation, $F_i : \mathbb{R}^d \to \mathbb{R}$ is a smooth non-convex function privately known by agent $i$. The notation $\mathbb{E}_{\xi_i}$ is the expected value of the random variable $\xi_i$ (e.g., random data samples) taken with respect to some local distribution. The above formulation is known as the consensus formulation since the agents share a common variable, which they need to agree upon [1]. We consider decentralized methods where the agents aim to find a solution of (1) through local interactions (each agent can only send and receive information to its immediate neighbors).

Two important measures of the performance of distributed (or decentralized) methods are the linear speedup and transient time. A decentralized method is said to achieve linear speedup if the gradient computational complexity needed to reach certain accuracy reduces linearly with the network size $n$. The transient time of a distributed method is the number of iterations needed to achieve linear speedup. A common method for solving problem (1) is the decentralized/distributed stochastic gradient descent (D$^2$GD) method [6], [7], where each agent employs a local stochastic gradient descent update and a local gossip step (there are several variations based on the order of the gossip step such as diffusion or consensus methods [7]–[9]). D$^2$GD is simple to implement; moreover, it has been shown to achieve linear speedup asymptotically [3]. This implies that the convergence rate of D$^2$GD asymptotically achieves the same network independent rate as the centralized (also known as parallel) stochastic gradient descent (PSGD) with a central coordinator. While being attractive, D$^2$GD suffers from an error or bias term caused by the heterogeneity between the local cost functions minimizers (e.g., heterogeneous data distributions across the agents) [8], [10]. The existence of such bias term will slow down the convergence of D$^2$GD, and hence enlarge its transient time.
Several bias-correction methods have been proposed to remove the bias of DSGD such as EXTRA [11], Exact-Diffusion (ED) (a.k.a. D^2 or NIDS) [12]–[15], and gradient-tracking (GT) methods [16]–[19]. Although these methods have been extensively studied, their convergence properties have not been fully understood as we now explain. Under convex stochastic settings, ED/D^2 is theoretically shown to improve upon the transient time of DSGD [20], [21], especially under sparse topologies. However, existing non-convex results imply that ED/D^2 has worse transient time compared to DSGD for sparse networks [15]. Moreover, the transient time of GT-methods are theoretically worse than DSGD under sparse networks even under convex settings [22]. These existing theoretical results imply that under non-convex settings, bias-correction methods can suffer from worse transient time compared to DSGD. However, empirical results suggest that both ED/D^2 and GT methods outperform DSGD under sparse topologies (without any acceleration) [15], [23]–[25]. This phenomenon is yet to be explained.

In this work, we provide a novel unified convergence analysis of several decentralized bias-correction methods including both ED/D^2 and GT methods under non-convex settings. We establish refined and improved convergence rate bounds over existing results. Moreover, our results show that bias-correction methods such as Exact-Diffusion/D^2 and GT methods have better network topology dependent bounds compared to DSGD. We also study these methods under the Polyak-Łojasiewicz (PL) condition [26], [27] and provide refined bounds over existing literature. Before we state our main contributions, we will go over the related works.

A. Related Works

There exists many works that study decentralized optimization methods under deterministic settings (full knowledge of gradients) – see [9], [11], [19], [28]–[33] and references therein. For example, the works [32]–[36] propose unified frameworks that cover several state-of-the-art methods and study their convergence, albeit under deterministic and convex settings. This work considers nonconvex costs and focuses on the stochastic learning setting where each agent has access to a random estimate of its gradient at each iteration. For this setting, DSGD is the most widely studied and understood method [3], [4], [37]–[48]. Under non-convex settings, the transient time of DSGD is on the order of $O(n^3/(1-\lambda)^4)$ [3], [4], [47], where $1-\lambda \in (0,1)$ denotes the network spectral gap that measures the connectivity of the network topology (e.g., it goes to zero for sparse networks). As a result, improving the convergence rate dependence on the network topology quantity $1-\lambda$ is crucial to enhance the transient time of decentralized methods.

The severe dependence on the network topology in DSGD is caused by the data heterogeneity between different agents [14], [47]. Consequently, the dependence on the network topology can be ameliorated by removing the bias caused by data heterogeneity. For example, the transient time of ED/D^2 has been shown to have enhanced dependence on the network topology compared to DSGD [20], [21] under convex settings. However, it is unclear whether bias-correction methods can achieve the same results for non-convex settings [15], [24], [49]–[52]. In fact, the established transient time of bias-correction methods such as ED/D^2 and GT in literature are even worse than that of DSGD. For instance, the best known transient time for both ED/D^2 and GT is on the order of $O(n^3/(1-\lambda)^6)$ [15], [24], which is worse than DSGD with transient time $O(n^3/(1-\lambda)^3)$. These counter-intuitive results naturally motivate us to study whether ED/D^2 and GT can enjoy an enhanced dependence on the network topology in the non-convex setting. It is also worth noting that the dependence on network topology established in existing GT references are worse than DSGD even for convex scenarios [22]. This work provides refined and enhanced convergence rates for both ED/D^2 and GT (as well as other methods such as EXTRA) under the non-convex setting.

In this work, we also study the convergence properties of decentralized methods under the Polyak-Łojasiewicz (PL) condition [26]. The PL condition can hold for non-convex costs, yet it can be used to establish similar convergence rates to strong-convexity rates [27]. For strongly-convex settings, the works [20], [21] showed that the transient time of ED/D^2 is on the order of $O(n^3/(1-\lambda)^3)$. These are the best available network bounds for decentralized methods so far for strongly-convex settings. It is still unclear, whether bias-correction methods can achieve similar bounds to DSGD under the PL condition. For example, the work [24] shows that under the PL condition, GT methods have transient time on the order $O(n^3/(1-\lambda)^3)$.

We remark that this work only considers non-accelerated decentralized methods with a single gossip round per iteration. It has been shown that combining GT methods with multiple gossip rounds can further improve the dependence on network topology [23], [53], and this technique can also be incorporated into our studied algorithm and its analysis. However, it is worth noting that the utilization of multiple gossip rounds in decentralized stochastic methods might suffer from several limitations. First, it requires the knowledge of the quantity $\lambda$ to decide the number of gossip rounds per iteration, which, however, might not be available in practice. Second, the multiple gossip rounds update can take even more time than a global average operation. For example, the experiments provided in [5, Table 17] indicate that, under a certain practical scenario, one gossip step requires half or third the communication overhead of a centralized RING-ALLREDUCE operation [54], which conducts global averaging. This implies that decentralized methods with as much as two or three gossip rounds per iteration can be more costly than global averaging. Third, the theoretical improvements brought by multiple gossip rounds rely heavily on gradient accumulation. Such gradient accumulation can easily cause large batch-size which are empirically and theoretically found to be harmful for generalization performance on unseen dataset [55], [56].

B. Main Contributions

Our main contributions are formally listed below.

- We unify the analysis of several well-known decentralized methods under non-convex and stochastic settings. In particular, we study the convergence properties of a general primal-dual algorithmic framework, called stochastic
We provide a novel analysis technique for these type of doubly stochastic matrices that are key to establish our refined convergence rates. We also study the convergence properties of matrices such as ED/D, EXTRA, and GT methods. In particular, we employ several novel transformations to SUDA that are key to establish our refined convergence rates. Our analysis provides improved network dependent bounds for the special cases of SUDA such as ED/D and GT methods compared to existing best known results. In addition, the established transient time of ED/D and ATC-GT have improved network topology dependence compared to DSGD – see Table I.

- We provide a novel analysis technique for these type of methods. In particular, we employ several novel transformations to SUDA that are key to establish our refined convergence rates bounds (see Remark 4). Our analysis provides improved network dependent bounds for the special cases of SUDA such as ED/D and GT methods compared to existing best known results. In addition, the established transient time of ED/D and ATC-GT have improved network topology dependence compared to DSGD – see Table I.

- We also study the convergence properties of SUDA under the PL condition. When specifying SUDA to ED/D, we achieve network dependent bound matching the best known bounds established under strongly-convex settings. When specifying SUDA to GT methods, we achieve an improved network dependent bound compared to current results even under strong-convexity – see Table II.

### TABLE I
Comparison with Existing Non-Convex Convergence Rates Highlighting the Network Quantities. Here, $\lambda^2 = \frac{1}{n} \sum_{i=1}^{n} \|\nabla f_i(0) - \nabla f(0)\|^2$ and $\lambda^2$ satisfies $\frac{1}{n} \sum_{i=1}^{n} \|\nabla f_i(x) - \nabla f(x)\|^2 \leq \lambda^2$ for all $x \in \mathbb{R}^d$ for DSGD. The quantity $\lambda = \rho(W - \frac{1}{n}11^T)$ is the mixing rate of the network where $W$ is the network combination matrix.

| Method | Work | Convergence Rate | Transient Time |
|--------|------|------------------|----------------|
| DSGD   | [47] | $O\left(\frac{1}{\sqrt{n}} + \frac{\lambda^{2/3}}{(1-\lambda)^{1/3}K^{2/3}} + \frac{\lambda^{2/3}}{(1-\lambda)^{2/3}K^{2/3}}\right)$ | $O\left(\frac{n^3}{(1-\lambda)^2}\right)$ |
| ED/D²  | [15] | $O\left(\frac{1}{\sqrt{n}} + \frac{n\lambda^2}{(1-\lambda)^2K} + \frac{n\lambda^2}{(1-\lambda)^3K^2}\right)$ | $O\left(\frac{n^3}{(1-\lambda)^2}\right)$ |
| This work | | $O\left(\frac{1}{\sqrt{n}} + \frac{n\lambda^2}{(1-\lambda)^2K} + \frac{n\lambda^2}{(1-\lambda)^3K^2}\right)$ | $O\left(\frac{n^3}{(1-\lambda)^2}\right)$ |
| ATC-GT | [24] | $O\left(\frac{1}{\sqrt{n}} + \frac{n\lambda^2}{(1-\lambda)^2K} + \frac{\lambda^2}{(1-\lambda)^3K^2}\right)$ | $O\left(\frac{n^3}{(1-\lambda)^2}\right)$ |
| This work | | $O\left(\frac{1}{\sqrt{n}} + \frac{n\lambda^2}{(1-\lambda)^2K} + \frac{\lambda^2}{(1-\lambda)^3K^2}\right)$ | $O\left(\frac{n^3}{(1-\lambda)^2}\right)$ |

### TABLE II
Comparison with Existing Network Dependent Transient Times under Both Strongly Convex and PL Condition Settings. Here, the quantity $\lambda = \rho(W - \frac{1}{n}11^T)$ is the mixing rate of the network where $W$ is the network combination matrix.

| Method | Work | Assumption | Transient Time |
|--------|------|------------|----------------|
| DSGD   | [47] | Strongly-convex | $O\left(\frac{n}{(1-\lambda)^2}\right)$ |
| ED/D²  | [20, 21] | Strongly-convex | $O\left(\frac{n}{(1-\lambda)^2}\right)$ |
| This work | PL condition | $O\left(\frac{n}{(1-\lambda)^2}\right)$ |
| ATC-GT | [22] | Strongly-convex | $O\left(\frac{n}{(1-\lambda)^2}\right)$ |
| [24] | PL condition | $O\left(\frac{n}{(1-\lambda)^2}\right)$ |
| This work | PL condition | $O\left(\max\left\{\frac{n}{(1-\lambda)^2}, \frac{1}{(1-\lambda)^3}\right\}\right)$ |

### Notation:
Vectors and scalars are denoted by lowercase letters. Matrices are denoted by uppercase letters. We use $\emf{col}\{a_1, \ldots, a_n\}$ (or $\emf{col}(a_i)_{i=1}^n$) to denote the vector that stacks the vectors (or scalars) $a_i$ on top of each other. We use $\emf{diag}\{d_1, \ldots, d_n\}$ (or $\emf{diag}(d_i)_{i=1}^n$) to denote a diagonal matrix with diagonal elements $d_i$. We also use $\emf{blkdiag}\{D_1, \ldots, D_n\}$ (or $\emf{blkdiag}(D_i)_{i=1}^n$) to denote a block diagonal matrix with block diagonal matrices $D_i$. The vector of all ones with size $n$ is denoted by $\emf{1}_n$, or $\emf{1}$ and size is known from context). The inner product of two vectors $a$ and $b$ is denoted by $\emf{a} \cdot \emf{b}$. The Kronecker product operation is denoted by $\otimes$. For a square matrix $A$, we let $\rho(A)$ denote the spectral radius of $A$, which is the largest absolute value of its eigenvalues. Upright bold symbols ($e.g., \emf{x}, \emf{f}, \emf{W}$) are used to denote augmented network quantities.

### II. General Algorithm Description
In this section, we describe the deterministic form of the studied algorithm and list several specific instances of interest to us.

#### A. General Algorithm
To describe the algorithm, we introduce the network quantities:

$$\emf{x} \triangleq \emf{col}\{x_1, \ldots, x_n\} \in \mathbb{R}^{dn}, \quad \emf{f}(\emf{x}) \triangleq \sum_{i=1}^{n} f_i(x_i).$$

(2)

We also introduce the matrix $\emf{B} \in \mathbb{R}^{dn \times dn}$ that satisfies

$$\emf{B} \emf{x} = \emf{0} \iff x_1 = x_2 = \cdots = x_n.$$  

(3)

Using the previous definitions, the general algorithmic framework can be described as follows. Set an arbitrary initial estimate $\emf{x}^0 \in \mathbb{R}^{dn}$ and set $\emf{y}^0 = \emf{0}$. Repeat for $k = 0, 1, \ldots$

$$\emf{x}^{k+1} = \emf{A} (\emf{C} \emf{x}^k - \alpha \emf{\nabla f}(\emf{x}^k)) - \emf{B} \emf{y}^k,$$

(4a)

$$\emf{y}^{k+1} = \emf{y}^k + \emf{B} \emf{x}^{k+1}.$$  

(4b)

Here, $\alpha > 0$ is the step size (learning rate), and the matrices $\emf{A} \in \mathbb{R}^{dn \times dn}$ and $\emf{C} \in \mathbb{R}^{dn \times dn}$ are doubly stochastic matrices.
that are chosen according to the network combination matrix introduced next.

B. Network Combination Matrix

We let \( W = [w_{ij}] \in \mathbb{R}^{n \times n} \) denote the network combination (weighting) matrix assumed to be symmetric. Here, the \((i, j)\)th entry \( w_{ij} \geq 0 \) is used by agent \( i \) to scale information received from agent \( j \). We consider a decentralized setup where \( w_{ij} = 0 \) if \( j \notin \mathcal{N}_i \), where \( \mathcal{N}_i \) is the neighborhood of agent \( i \). If we introduce the augmented combination matrix \( W = W \otimes I_d \in \mathbb{R}^{dn \times dn} \), then the matrices \( A, B, C \) can be chosen as a function of \( W \) to recover several existing decentralized methods. Note that if \( u = \text{col}\{w_{ik}\}_{k=1}^n \) where \( u_i \in \mathbb{R}^d \) is local to agent \( i \), then, the \( i \)th block of \( Wu = \text{col}\{\sum_{j \in \mathcal{N}_i} w_{ij}u_j\}_{j=1}^n \) can be computed by agent \( i \) through local interactions with its neighbors.

C. Relation to Existing Decentralized Methods

Below, we list several important well-known decentralized algorithms that are covered in our framework. Please see Appendix A for more details.

**Exact-Diffusion/D\(^2\):** If we select \( A = W, B = (I - W)^{1/2}, \) and \( C = I \). Then, algorithm (4) becomes equivalent to ED/D\(^2\) [12, 15]:

\[
x^{k+2} = \sum_{j \in \mathcal{N}_i} w_{ij} \left( 2x^{k+1} - x^k - \alpha (\nabla f_j(x^{k+1}) - \nabla f_j(x^k)) \right),
\]

with \( x_0 = \sum_{j \in \mathcal{N}_i} w_{ij} \left( x^0 - \alpha \nabla f_j(x^0) \right) \).

**EXTRA:** If we instead select \( A = I, B = (I - W)^{1/2}, \) and \( C = W \), then algorithm (4) is equivalent to EXTRA [11].

**Gradient-Tracking (GT) methods:** Consider the adapt-then-combine gradient-tracking (ATC-GT) method [16]:

\[
x^{k+1} = \sum_{j \in \mathcal{N}_i} w_{ij} \left( x^k - \alpha g^j \right)
\]

\[
g^{k+1} = \sum_{j \in \mathcal{N}_i} w_{ij} \left( g^k + \nabla f_j(x^{k+1}) - \nabla f_j(x^k) \right).
\]

With proper initialization, the above is equivalent to (4) when \( A = W^2, B = I - W, \) and \( C = I \). We can also recover other gradient-tracking variants such as the non-ATC-GT method [18] where the communication (gossip) step only involves the terms \( x^k \) and \( g^k \) in the update of each vector. Similarly, we can also cover the semi-ATC-GT variations [17] where only the update of \( x^k \) or \( g^k \) uses the ATC structure.

Remark 1: (Relation with other frameworks). The unified decentralized algorithm (UDA) from [32] is equivalent to (4) if \( A \) and \( B^2 \) commute (i.e., \( AB^2 = B^2A \)). Therefore, all the methods covered in [32] are also covered by our framework (such as DLM [57]). Moreover, under certain conditions, the frameworks from [33] and [34] can also be related with (4) – see supp. material or arXiv version [58, Appendix A]. The works [32]–[34] only studied convergence under deterministic and convex settings. In contrast, we study the non-convex and stochastic case, and more importantly, we establish tighter rates for the above special bias-correction methods, which is the main focus of this work.

III. Stochastic UDA and Assumptions

In this section, we describe the stochastic version of algorithm (4) and list the assumptions used to analyze it.

As stated in problem (1), we consider stochastic settings where each agent may only have access to a stochastic gradient \( \nabla F_i(x_i, \xi^k) \) at each iteration \( k \) instead of the true gradient. This scenario arises in online learning settings, where the data are not known in advance; hence, we do not have access to the actual gradient. Moreover, even if all the data is available, the true gradient might be expensive to compute for large datasets and can be replaced by a gradient at one sample or a mini-batch.

Replacing the actual gradient by its stochastic approximation in (4), we get the Stochastic Unified Decentralized Algorithm (SUADA):

\[
x^{k+1} = \Lambda \left( Cx^k - \alpha \nabla F(x^k, \xi^k) \right) - By^k,
\]

\[
y^{k+1} = y^k + Bx^{k+1},
\]

where \( \nabla F(x, \xi^k) = \text{col}\{\nabla F_1(x_1, \xi^k), \ldots, \nabla F_n(x_n, \xi^k)\} \).

We next list the assumptions used in our analyses. Our first assumption is about the network combination matrix given next.

**Assumption 1 (Combination matrix):** The combination matrix \( W \) is assumed to be doubly stochastic, symmetric, and primitive. Moreover, we assume that the matrices \( A, B^2, C \) are chosen as a polynomial function of \( W \):

\[
A = \sum_{l=0}^p a_l W^l, \quad B^2 = \sum_{l=0}^p b_l W^l, \quad C = \sum_{l=0}^p c_l W^l,
\]

where \( p \geq 1 \). The constants \( \{a_l, b_l, c_l\}_{l=0}^p \) are chosen such that \( A \) and \( C \) are doubly stochastic and the matrix \( B \) satisfies (3).

Under Assumption 1, the combination matrix \( W \) has a single eigenvalue at one, denoted by \( \lambda_1 = 1 \). Moreover, all other eigenvalues, denoted by \( \{\lambda_i\}_{i=2}^n \), are strictly less than one in magnitude [40], and the mixing rate of the network is:

\[
\lambda \triangleq \rho \left( W - \frac{1}{n} 1 1^T \right) = \max_{i \in [2, \ldots, n]} |\lambda_i| < 1.
\]

Note that the assumptions on \( A, B^2, C \) are mild and hold for all the special cases described before.

We now introduce the main assumption on the objective function.

**Assumption 2 (Objective function):** Each function \( f_i : \mathbb{R}^d \rightarrow \mathbb{R} \) is \( L \)-smooth:

\[
\|\nabla f_i(z) - \nabla f_i(y)\| \leq L\|z - y\|, \quad \forall z, y \in \mathbb{R}^d,
\]

for some \( L > 0 \). We also assume that the aggregate function \( f(x) = \frac{1}{n} \sum_{i=1}^n f_i(x) \) is bounded below, i.e., \( f(x) \geq f^* > -\infty \quad \forall x \in \mathbb{R}^d \), where \( f^* \) denotes the optimal value of \( f \).

The above assumption is standard to establish convergence under non-convex settings. Note that we do not impose the strong assumption of bounded gradient dissimilarity, which is required to establish convergence of DSGD – see [3], [4], [47].

We next list our assumption on the stochastic gradient. To do that, we define the filtration generated by the random process (7):

\[ \mathcal{F}^k \triangleq \{x^0, x^2, \ldots, x^k\} \]. The filtration \( \mathcal{F}^k \) can be interpreted as the collection of all information available on the past iterates up to time \( k \).
Assumption 3 (Gradient noise): For all \( \{i\}_{i=1}^{n} \) and \( k = 0, 1, \ldots, \) we assume that the following holds
\[
\begin{align*}
\mathbb{E} \left[ \nabla F_i(x_i^k, \xi_i^k) - \nabla f_i(x_i^k) \right] &\mathcal{F}^k = 0, \quad (11a) \\
\mathbb{E} \left[ \nabla F_i(x_i^k, \xi_i^k) - \nabla f_i(x_i^k) \right]^2 &\mathcal{F}^k \leq \sigma^2, \quad (11b)
\end{align*}
\]
for some \( \sigma^2 > 0 \). We also assume that conditioned on \( \mathcal{F}^k \), the random data \( \{\xi_i^k\} \) are independent of each other for all \( \{i\}_{i=1}^{n} \) and \( \{k\}_{k=0}^{\infty} \).

The previous assumptions will be used to analyze SUDA (7) for general non-convex costs. In the sequel, we will also study SUDA under the following additional assumption.

Assumption 4 (PL condition): The aggregate function \( f(x) = \frac{1}{n} \sum_{i=1}^{n} f_i(x) \) satisfies the PL inequality:
\[
2\mu (f(x) - f^*) \leq \| \nabla f(x) \|^2, \quad \forall x \in \mathbb{R}^d,
\]
for some \( \mu > 0 \) where \( f^* \) denote the optimal value of \( f \).

The above condition indicates that every stationary point is a global minimizer, which is weaker than many assumptions used to establish linear convergence without strong-convexity [27]. Note that the PL condition is also referred to as the gradient dominated condition [59].

IV. FUNDAMENTAL TRANSFORMATIONS

The updates of SUDA (7), while useful for implementation purposes, are not helpful for analysis purposes. In this section, we will transform SUDA (7) into an equivalent recursion that is fundamental to arrive at our results.

A. Transformation I

If we introduce the quantities
\[
\begin{align*}
\tilde{x}^k &\triangleq y^k - B \tilde{x}^k, \quad (13a) \\
\check{x}^k &\triangleq \frac{1}{n} (I_n \otimes I_d) \check{x}^k = \frac{1}{n} \sum_{i=1}^{n} x_i^k, \quad (13b) \\
\hat{x}^k &\triangleq 1_n \otimes \check{x}^k, \quad (13c) \\
\hat{s}^k &\triangleq B z^k + \alpha A \nabla f(\hat{x}^k), \quad (13d)
\end{align*}
\]
then algorithm (7) can be rewritten as (see supp. material):
\[
x^{k+1} = (A C - B^2) x^k - s^k - \alpha A (L \nabla f(\hat{x}^k) - \nabla f(\tilde{x}^k) + w^k) \tag{14a}
\]
where \( w^k \) is the gradient noise defined as
\[
w^k \triangleq \nabla F(\hat{x}^k, \xi^k) - \nabla f(\hat{x}^k). \tag{15}
\]

Remark 2: (MOTIVATION BEHIND \( s^k \)). Suppose that \( (x, s) \) is a fixed point of the deterministic form of (14) (gradient noise \( w^k = 0 \)) where \( s \in \mathbb{R}^{dn} \) and \( x = (x_1, \ldots, x_n) \in \mathbb{R}^{dn} \). Then, from (14b) it holds that \( 0 = B^2 x \iff x_1 = \cdots = x_n = x \) and from (14a), we have \( s = 0 \). Therefore, using the definition of \( s^k \) in (13d) and \( x = 1 \otimes x \), it follows that there exists some \( z \in \mathbb{R}^{dn} \) such that
\[
s = \alpha A \nabla f(\hat{x}) + B z = 0
\]
\[
\Rightarrow \frac{1}{n} (I_n \otimes I_n) (\alpha A \nabla f(\hat{x}) + B z) = \frac{\alpha}{n} \sum_{i=1}^{n} \nabla f_i(x). = 0.
\]
Hence, \( x \) is a stationary point of problem (1).

B. Transformation II

We next exploit the structure of the matrices \( A, B, \) and \( C \) to further transform recursion (14) into a more useful form. Under Assumption 1, the combination matrix \( W \) can be decomposed as
\[
W = U \hat{A} U^T = \left[ \frac{1}{\sqrt{n}} 1 \hat{U} \right] \left[ \begin{array}{cc} I_d & 0 \\ 0 & \tilde{A} \end{array} \right] \left[ \frac{1}{\sqrt{n}} 1 \hat{U}^T \right],
\]
where \( \hat{A} \triangleq \hat{A} \otimes I_d \in \mathbb{R}^{d(n-1)\times d(n-1)} \), \( U \in \mathbb{R}^{dn \times dn} \) is an orthogonal matrix, and \( \hat{U} \triangleq \hat{U} \otimes I_d \in \mathbb{R}^{dn \times d(n-1)} \) satisfies:
\[
\hat{U}^T U = I, \quad \hat{U}^T \hat{U} = I - \frac{1}{n} 1 1^T \otimes I_d,
\]
and \( (1^T \otimes I_d) U = 0 \). Since \( A, B^2, C \) are chosen as polynomial function of \( W \) as described in Assumption 1, it holds that
\[
A = \left[ \frac{1}{\sqrt{n}} 1 \otimes I_d \right] \hat{U} \left[ \begin{array}{cc} I_d & 0 \\ 0 & \tilde{A} \end{array} \right] \left[ \frac{1}{\sqrt{n}} 1 \hat{U}^T \right], \tag{17a}
\]
\[
C = \left[ \frac{1}{\sqrt{n}} 1 \otimes I_d \right] \hat{U} \left[ \begin{array}{cc} I_d & 0 \\ 0 & \tilde{A} \end{array} \right] \left[ \frac{1}{\sqrt{n}} 1 \hat{U}^T \right], \tag{17b}
\]
\[
B^2 = \left[ \frac{1}{\sqrt{n}} 1 \otimes I_d \right] \hat{U} \left[ \begin{array}{cc} 0 & 0 \\ 0 & \tilde{A}^2 \end{array} \right] \left[ \frac{1}{\sqrt{n}} 1 \hat{U}^T \right], \tag{17c}
\]
where \( \tilde{A}_a = \text{diag} \{\lambda_{ai}\}_{i=2}^{n} \otimes I_d \), \( \tilde{A}_b = \text{diag} \{\lambda_{bi}\}_{i=2}^{n} \otimes I_d \), and \( \tilde{A}_c = \text{diag} \{\lambda_{ci}\}_{i=2}^{n} \otimes I_d \) with \( \tilde{A}_a \triangleq \sum_{i=0}^{n} a_i \lambda_i^2, \tilde{A}_b \triangleq \sum_{i=0}^{n} b_i \lambda_i^2, \) and \( \tilde{A}_c \triangleq \sum_{i=0}^{n} c_i \lambda_i^2 \). Moreover, \( \tilde{A}_b \) is positive definite due to the null space condition (3). Note that
\[
(1^T \otimes I_d) s^{k+1} = (1^T \otimes I_d) (B z^k + \alpha A \nabla f(\hat{x}^k)) = \alpha \sum_{i=1}^{n} \nabla f_i(\hat{x}^k). \]
Moreover, utilizing the structure of \( U \), we have
\[
\begin{align*}
U^T x^k &\triangleq \left[ \frac{\sqrt{n}}{n} \hat{x}^k \right], \\
U^T s^k &\triangleq \left[ \frac{\sqrt{n}}{n} \alpha \nabla f(\hat{x}^k) \right] \tag{18a}
\end{align*}
\]
\[
U^T \nabla f(x) = \left[ \frac{\sqrt{n}}{n} \nabla f(\hat{x}^k) \right], \\
U^T w^k = \left[ \frac{\sqrt{n}}{n} w^k \right] \tag{18b}
\]
where
\[
\nabla f(\hat{x}^k) \triangleq \frac{1}{n} (1^T \otimes I_n) \nabla f(\hat{x}^k) = \frac{1}{n} \sum_{i=1}^{n} \nabla f_i(x_i^k). \tag{19a}
\]
\[
\nabla f(\hat{x}^k) = \frac{1}{n} \sum_{i=1}^{n} (\nabla F_i(x_i^k, \xi_i^k) - \nabla f_i(x_i^k)). \tag{19b}
\]
Therefore, multiplying both sides of (14) by \( U^T \), using the previous quantities and the structure of \( \Lambda_a, \Lambda_b, \Lambda_c \) given in (17), we find that (see supp. material)

\[
\tilde{x}^{k+1} = \tilde{x}^k - \alpha \nabla \tilde{V}(x^k) - \alpha w^k, \tag{20a}
\]

\[
\begin{bmatrix}
U^T x^{k+1} \\
\Lambda_b^{-1} U^T s^{k+1}
\end{bmatrix} = G \begin{bmatrix}
U^T x^k \\
\Lambda_b^{-1} U^T s^k
\end{bmatrix} - \alpha \begin{bmatrix}
\Lambda_a U^T \nabla f(\tilde{x}^k) - \nabla \tilde{V}(x^k) + w^k \\
\Lambda_a U^T \nabla f(\tilde{x}^{k+1})
\end{bmatrix}, \tag{20b}
\]

where

\[
G = \begin{bmatrix}
\Lambda_a \hat{\Lambda}_e - \hat{\Lambda}_b \\
\Lambda_b
\end{bmatrix} \in \mathbb{R}^{2d(n-1) \times 2d(n-1)}. \tag{21}
\]

The convergence of (20) is governed by the matrix \( G \). We next introduce a fundamental factorization of \( G \) that will be used to transform (20) into our final key recursion. The next result is proven in Appendix B1.

**Lemma 1 (Fundamental factorization):** Suppose that the eigenvalues of \( G \) are strictly less than one in magnitude, then, there exists an invertible matrix \( \hat{V} \) such that the matrix \( G \) admits the similarity transformation

\[
G = \hat{V} \Gamma \hat{V}^{-1}, \tag{22}
\]

where \( \Gamma \) satisfies \(|\Gamma| < 1\). □

For the convergence of (20), it is necessary that \( G \) is a stable matrix (has eigenvalues strictly less than one in magnitude). To see this suppose that \( \nabla f(x) = w = 0 \), then the convergence is dictated by (20b), which diverges if unstable \( G \) if \( \tilde{U}^T x^0 \neq 0 \) or \( \tilde{U}^T s^0 \neq 0 \). Thus, we implicitly assume that \( G \) is a stable matrix. Explicit expressions for \( \Gamma \) and \( \hat{V} \) for ED/D², EXTRA, and GT methods are derived in Appendix B2, where we also find exact expressions for the eigenvalues of \( G \) for these methods.

Finally, multiplying both sides of (20b) by \( \frac{1}{v} \hat{V}^{-1} \) \((v > 0)\) and using the structure (22), we arrive at the following key result.

**Lemma 2 (Final transformed recursion):** Under Assumption 1, there exists an invertible matrix \( \hat{V} \) such that the recursion (20) can transformed into

\[
\tilde{x}^{k+1} = \tilde{x}^k - \alpha \nabla \tilde{V}(x^k) - \alpha w^k, \tag{23a}
\]

\[
\hat{e}^{k+1} = \hat{v}^{k+1} = \Gamma \hat{e}^k - \alpha \hat{V}^{-1} \left[ \frac{1}{v} \hat{A}_a \hat{U}^T \nabla f(\tilde{x}^k) - \nabla \tilde{V}(x^k) + w^k \right], \tag{23b}
\]

where \( \Gamma \) was introduced in (22), \( v \) is an arbitrary strictly positive constant, and

\[
\hat{e}^k = \frac{1}{v} \hat{V}^{-1} \begin{bmatrix}
\hat{U}^T x^k \\
\hat{A}_b^{-1} \hat{U}^T s^k
\end{bmatrix}. \tag{24}
\]

Remark 3: (Deviation from average). Recall that \( \tilde{x}^k = 1 \otimes \tilde{x}^k \) where \( \tilde{x}^k = \frac{1}{n} \sum_{i=1}^n x^i \). Since \( \tilde{U}^T \tilde{U} = I \), it holds that

\[
||\tilde{U}^T x||^2 = x^T \tilde{U}^T \tilde{U}^T \tilde{U}^T x = ||\tilde{U}^T x^k||^2 \overset{(16)}{=} ||x^k - \tilde{x}||^2.
\]

Therefore, the quantity \( ||\tilde{U}^T x||^2 \) measures the deviation of \( x^k \) from the average \( \tilde{x}^k \). Similarly, \( ||\tilde{U}^T s||^2 \) measures the deviation of \( s^k \) from the average \( \frac{1}{n} \sum_{i=1}^n \nabla f_i(\tilde{x}^k) \) (see (13d)). Now, using (24), we have

\[
||v \hat{v} \tilde{e}^k||^2 = ||\tilde{U}^T x^k||^2 + ||\hat{A}_b^{-1} \hat{U}^T s^k||^2. \tag{25}
\]

Thus, the vector \( e^k \) can be interpreted as a measure of a weighted deviation of \( x^k \) and \( s^k \) from \( \tilde{x}^k \) and \( 1 \otimes \frac{1}{n} \sum_{i=1}^n \nabla f_i(\tilde{x}^k) \), respectively.

**Remark 4:** This work handles the deviation of the individual vectors from the average \( x^k - \tilde{x}^k \), and the deviation of the “gradient-tracking” variable \( s^k \) from the averaged-gradients \( 1 \otimes \frac{1}{n} \sum_{i=1}^n \nabla f_i(\tilde{x}^k) \) as one augmented quantity. This leads to two coupled error terms given by (20). This is one main reason that allows us to obtain tighter network-dependent rates compared to previous works. The rigorous factorization of the matrix \( G \) leads us to arrive at the final transformed recursion (23) with contractive matrix \( \Gamma \), which is key for our result.

This is in contrast to other works. For example, in previous GT works (see e.g., [18], [22], [24]), the deviation of the individual vectors from the average \( x^k - \tilde{x}^k \), and the deviation of the gradient-tracking variable from the averaged-gradients are handled independently, and thus, they do not exploit the coupling matrix between these two network quantities. □

V. CONVERGENCE RESULTS

In this section, we state and discuss the convergence results.

A. Convergence Under General Non-Convex Costs

The following result establishes the convergence under general non-convex smooth costs.

**Theorem 1 (Convergence of SUDA):** Suppose that Assumptions 1–3 hold and the step size satisfies \( \alpha \leq \min \left\{ \frac{1}{2K}, \frac{1-\gamma}{2L_{uv12} \lambda_a}, \frac{\sqrt{\lambda_a(1-\gamma)}}{2L_{uv12} \lambda_a} \right\} \). Then, the iterates \( \{x^k\} \) of SUDA with \( x^0 = 1 \otimes x^0 (x^0 \in \mathbb{R}^d) \) satisfy

\[
\begin{alignat}{2}
\frac{1}{K} \sum_{k=0}^{K-1} & \left[ \frac{E \| \nabla f(x^k) \|^2}{\alpha K} + E \| \nabla \tilde{V}(x^k) \|^2 \right] \\
& \leq \frac{8(f(x^0) - f^*)}{\alpha K} + \frac{4aL \sigma^2}{n} + \frac{12\alpha^2 L^2 v_1^2 v_2^2 \zeta_0^2}{\alpha K} \\
& + \frac{16\alpha^2 L^2 v_1^2 v_2^2 \lambda_a \sigma^2}{1-\gamma} + 16\alpha^2 L^4 v_1^2 v_2^2 \lambda_a \sigma^2 \frac{\lambda_a}{\lambda_a^2(1-\gamma)^2 n}, \tag{26}
\end{alignat}
\]

where \( v_1 \triangleq \| \hat{V} \|, \; v_2 \triangleq \| \hat{V}^{-1} \|, \; \gamma \triangleq \| \Gamma \| < 1, \; \lambda_a \triangleq \| \Lambda_a \|, \) and

\[
\zeta_0 \triangleq \frac{1}{n} \left[ \left( A - \frac{1}{n} I_d \otimes I_d \right) \left( \nabla f(x^0) - 1 \otimes \nabla f(x^0) \right) \right].
\]
Consequently, if we set \( \alpha = \frac{1}{2 \beta \lambda + \sqrt{K/n}} \) where \( \beta \triangleq 1 + \varepsilon_{k} \varepsilon_{r}^{2} \frac{1}{1 - \gamma} + \varepsilon_{k} \varepsilon_{r}^{2} \frac{1}{\Delta_{o}(1 - \gamma)} \). Then, we obtain the convergence rate
\[
\frac{1}{K} \sum_{k=0}^{K-1} E[\|\nabla f(x^{k})\|^2] \leq O\left(\frac{\sigma}{\sqrt{nK}} + \frac{1}{K} + \frac{\sigma^2 K}{n + \sigma^2 K} + \frac{n \zeta_0^2}{nK + \sigma^2 K^2}\right).
\] (27)

The proof of Theorem 1 is established in Appendix C. The convergence rate (27) shows that SUDA enjoys linear speedup since the dominating term is \(O(1/\sqrt{nK})\) for sufficiently large \(K\) [3]. Note that the rate given in (27) treats the network quantities \(\{\gamma, \lambda_o, \lambda_0, v_1, v_2\}\) as constants. However, these quantities can have a significant influence on the convergence rate as explained in the introduction. While the above result holds for EXTRA, ED/D^2 and GT methods, it is still unclear whether these methods can achieve enhanced transient time compared to DSGD. To show the network affect and see the implication of Theorem 1 in terms of network quantities, we will specialize Theorem 1 to ED/D^2 (5) and ATC-GT (6) and reflect the values of the \(\{\gamma, \lambda_o, \lambda_0, v_1, v_2\}\) in the convergence rate.

**Corollary 1 (ED/D^2 convergence):** Suppose that all the conditions given in Theorem 1 hold and assume further that \(W\) is positive semi-definite. If we set \(\alpha = \sqrt{n/K}\) and \(K \geq \max\left\{4nL^2, \frac{32L^2 \sigma_n^2}{(1 - \gamma)^2}, \frac{\sqrt{2} \sigma L \mu}{\sqrt{(1 - \lambda)^3/2}}\right\}\), then ED/D^2 (5) with \(x^0 = 1 \otimes x^0(x^0 \in \mathbb{R}^d)\) has convergence rate
\[
\frac{1}{K} \sum_{k=0}^{K-1} E[\|\nabla f(x^{k})\|^2] \leq O\left(\frac{f(x^{0}) - f^*}{\sqrt{nK}} + \frac{\sigma^2}{\sqrt{nK}} + \gamma^\alpha \frac{n \lambda^2}{(1 - \lambda) K^2} + \frac{n \lambda^2}{(1 - \lambda)^3 K^2} + \frac{n \lambda^2 \zeta_0^2}{(1 - \lambda)^2 K^2}\right),
\] (28)

where \(\alpha\) is the minimum non-zero eigenvalue of \(W\). Moreover, we have \(\zeta_0^2 = \frac{1}{n} \sum_{i=1}^{n} \left\|\sum_{j \in N_i} w_{ij} \nabla f_j(x^0) - \nabla f(x^0)\right\|^2 \leq \lambda^2 \zeta_0^2\), where \(\zeta_0^2 = \lambda^2 \sum_{i=1}^{n} \left\|\nabla f_i(x^0) - \nabla f(x^0)\right\|^2\).

**Proof:** The proof follows by substituting the bounds (44) established in Appendix B into (26).

**Corollary 2 (ATC-GT convergence):** Suppose that all the conditions given in Theorem 1 hold and assume further that \(W\) is positive semi-definite. If we set \(\alpha = \sqrt{n/K}\) and \(K \geq \max\left\{4nL^2, \frac{32L^2 \sigma_n^2}{(1 - \gamma)^2}, \frac{\sqrt{2} \sigma L \mu}{\sqrt{(1 - \gamma)^3/2}}\right\}\), then ATC-GT (6) with \(x^0 = 1 \otimes x^0(x^0 \in \mathbb{R}^d)\) has convergence rate
\[
\frac{1}{K} \sum_{k=0}^{K-1} E[\|\nabla f(x^{k})\|^2] \leq O\left(\frac{f(x^{0}) - f^*}{\sqrt{nK}} + \frac{\sigma^2}{\sqrt{nK}} + \gamma^\alpha \frac{n \lambda^2}{(1 - \lambda) K^2} + \frac{n \lambda^2}{(1 - \lambda)^3 K^2} + \frac{n \lambda^2 \zeta_0^2}{(1 - \lambda)^2 K^2}\right),
\] (29)

where \(\zeta_0^2 = \frac{1}{n} \sum_{i=1}^{n} \left\|\sum_{j \in N_i} w_{ij} \nabla f_j(x^0) - \nabla f(x^0)\right\|^2 \leq \lambda^2 \zeta_0^2\), and \(\zeta_0^2 = \lambda^2 \sum_{i=1}^{n} \left\|\nabla f_i(x^0) - \nabla f(x^0)\right\|^2\).

**Proof:** The proof follows by substituting the bounds (45) established in Appendix B2 into Theorem 1.

Note that linear speedup is achieved when the dominating term is \(O\left(\frac{1}{\sqrt{nK}}\right)\). This is case when \(K\) is sufficiently large so that the higher order terms are less than or equal to \(O\left(\frac{1}{\sqrt{nK}} B_i(g)\right)\). For example, for ED/D^2 bound in (28), linear speedup is achieved when
\[
O\left(\frac{n \lambda^2}{(1 - \lambda) K^2} + \frac{n \lambda^2}{(1 - \lambda)^3 K^2} + \frac{n}{(1 - \lambda)^2 K^2}\right) \leq O\left(\frac{1}{\sqrt{nK}}\right).
\]

The above holds when \(K \geq O(n^{3/2}/(1 - \lambda)^2)\). Here, we treated \(\Delta\) for ED/D^2 as constant since, for example, if we set \(W = (1 - \theta)W + \theta I\) for constant \(\theta > 0\), then it holds that \(\Delta \in (\theta, 1)\). Table I compares the our results with existing works. It is clear that our bounds are tighter in terms of the spectral gap \(1 - \lambda\). Moreover, ED/D^2 and GT methods have enhanced transient time compared to DSGD.

**Remark 5 (Step size selection):** In Corollaries 1 and 2, the step size is chosen as \(\alpha = \sqrt{n/K}\) to simplify the expressions. This choice is not optimal and tighter rates can be obtained if we meticulously select the step size. For example, we can select the step size as in Theorem 1 and obtain a rate similar to (27) where the dominating term is \(O(\sigma/\sqrt{nK})\) instead of \(O(\sigma^2/\sqrt{nK})\). We can even get a tighter rate by carefully selecting the step size similar to [47]. However, such choices does not affect the transient time order in terms of network quantities, which is the main conclusion of our results.

**Remark 6:** (EXTRA and Other GT Variations). We remark that the matrix \(G\) defined in (21) is identical for both EXTRA (39) and ED/D^2. Hence, the convergence rate of ED/D^2 given in (28) also holds for EXTRA with the exception of the value of \(\lambda^2\) in the numerators, which should be replaced by one for EXTRA (\(\lambda^2 \rightarrow 1\) in numerators). Likewise, for the non-ATC-GT and semi-ATC-GT (see Appendix A), the matrix \(G\) is identical to ATC-GT and the convergence rate of ATC-GT given (29) holds for these other variations except for the value of \(\lambda^2\) in the numerators, which is one for non-ATC-GT (\(\lambda^2 \rightarrow 1\) in numerators) and \(\lambda\) for the semi-ATC-GT (\(\lambda^2 \rightarrow \lambda\) in numerators). Please see Appendix B2 for more details.

Finally, note that for static and undirected graphs, our technique can be used to improve the network bounds for EXTRA, ED/D^2, and GT modifications for other cost-function settings such variance-reduced settings [60].

**B. Convergence Under PL Condition**

We next state the convergence of SUDA under the PL condition given in Assumption 4.

**Theorem 2 (PL case):** Suppose Assumptions 1–4 hold and that \(\alpha \leq \min\left\{\frac{1 - \gamma}{2L}, \frac{1 - \gamma}{\sqrt{2(1 - \lambda)^2}}, \frac{\mu \lambda^2 (1 - \gamma)^{1/3}}{2L(1 - \gamma)^2}\right\}\). The itersates \(\{x^k\}\) of SUDA with \(x^0 = 1 \otimes x^0(x^0 \in \mathbb{R}^d)\) satisfy
\[
\frac{1}{n} \sum_{i=1}^{n} E[|f(x^k) - f^*|] \leq \left(1 - \frac{\alpha \mu}{2}\right)^k r_0
\]
\[
+ O\left(\frac{\alpha L \sigma^2}{\mu} + \frac{\alpha^2 L^2 \varepsilon_1^2 \varepsilon_2^2 \lambda^2 \sigma^2}{\mu (1 - \gamma)} + \frac{n \lambda^2}{(1 - \gamma)^2 n}\right)
\] (30)
where $r_0 = 2\mathbb{E}[f(x_0) - f^\star] + \alpha^2 L^2 \sum_{i,j} (\frac{\gamma_d^2}{\lambda_d^2})^2$ and quantities $v_1, v_2, \gamma, \lambda_d, \lambda_0$ and $\gamma_0$ are defined as in Theorem 1.

Theorem 2 is proven in the supp. material. To discuss the implication of the above result, we specialize it to ED/D and ATC-GT. The proof of the next results are trivial and given in the supplemental material.

**Corollary 3 (ED/D$^2$ convergence under PL condition):** Let the same conditions as in Theorem 2 hold and further that $W$ is positive semi-definite. If the step size satisfies

$$\alpha \leq \min \left\{ \frac{1-\sqrt{3}}{3}, \frac{\sqrt{1+3\lambda}}{2L^2}, \frac{(1-\sqrt{3})\lambda}{\sqrt{27L^2}}, \left(\frac{(1-\sqrt{3})(1-\sqrt{3})\lambda}{18L^2}\right)^{1/3} \right\},$$

then

$$\text{ED/D}^2 (5) \text{ with } x^0 = 1 \odot x_0 \text{ converges as}
\begin{align*}
\frac{1}{n} \sum_{i=1}^{n} E[f(x_i^h) - f^\star] \leq & \left(1 - \frac{\alpha \mu}{2}\right) r_0 + O\left(\frac{\alpha L \sigma^2}{\mu n} + \frac{\alpha^2 L^2 \lambda^2 \sigma^2}{\mu(1-\lambda)} + \frac{\alpha^4 L^4 \lambda^2 \sigma^2}{\mu(1-\lambda)^3 n}\right),
\end{align*}

\begin{align*}
& \quad + O\left(\frac{\lambda^2 \sigma^2}{K^2} + \frac{\zeta^2}{K^4(1-\lambda)^2} + \frac{\lambda^2 \sigma^2}{K^4(1-\lambda)^3 n}\right),
\end{align*}

(32)

where $O(\cdot)$ hides logarithmic factors.

**Corollary 4 (ATC-GT convergence under PL condition):** Let the same conditions as in Theorem 2 hold and assume further that $W$ is positive semi-definite. Then, ATC-GT (6) with $x^0 = 1 \odot x_0 \in \mathbb{R}^d$ and $\alpha \leq \min \left\{ \frac{1-\lambda}{6L^2}, \frac{1-\lambda}{\sqrt{18L^2}}, \left(\frac{(1-\sqrt{3})\lambda}{432L^2}\right)^{1/3} \right\}$ has convergence rate:

$$\frac{1}{n} \sum_{i=1}^{n} E[f(x_i^h) - f^\star] \leq \left(1 - \frac{\alpha \mu}{2}\right) r_0 + O\left(\frac{\alpha L \sigma^2}{\mu n} + \frac{\alpha^2 L^2 \lambda^2 \sigma^2}{\mu(1-\lambda)} + \frac{\alpha^4 L^4 \lambda^2 \sigma^2}{\mu(1-\lambda)^3 n}\right),$$

(33)

where $r_0 = 2\mathbb{E}[f(x^0) - f^\star] + 2\alpha \sigma^2 + \frac{2\alpha \sigma^2}{(1-\lambda)^2}$ and $\zeta^2 = \frac{2\alpha \sigma^2}{n} \sum_{i,j=1} E[(\frac{\lambda_d^2}{\lambda_0^2})]$. Hence, if we set $\alpha = 2 \ln(K^2)/\mu K$, then it holds that

$$\frac{1}{n} \sum_{i=1}^{n} E[f(x_i^h) - f^\star] \leq \left(1 - \frac{\alpha \mu}{2}\right) r_0$$

$$+ O\left(\frac{\lambda^2 \sigma^2}{K^2(1-\lambda)} + \frac{\zeta^2}{K^4(1-\lambda)^2} + \frac{\lambda^2 \sigma^2}{K^4(1-\lambda)^3 n}\right).$$

(34)

where $O(\cdot)$ hides logarithmic factors.

We note that as in Remark 5, the selected step sizes in Corollaries 3 and 4 are not optimized. The hidden log factors can be removed if we adopt decaying step-sizes techniques (e.g., [43]). However, the main conclusion we want to emphasize is the network dependent bounds, which do not change if select better step size choices. Under the PL condition, linear speedup is achieved when $K$ is large enough such that the dominating term is $O(\frac{1}{\sqrt{\mu n}})$. Table II compares the transient times implied by our result compared to existing results. Our results significantly improves upon existing GT results. Moreover, our bound for ED/D$^2$ matches the existing bound from [20], [21], which are under the stronger assumption of strong-convexity.

**Remark 7:** (STEADY-STATE ERROR). For constant step size $\alpha$ independent of $k$, we can let $k$ go to $\infty$ in (31) and (33) to arrive at the following steady state results (lim $\sum_{k=1}^{\infty} \frac{1}{n} \sum_{i=1}^{n} E[f(x_i^k) - f^\star]$).

- For ED/D$^2$, we have

$$O\left(\frac{\alpha L \sigma^2}{\mu n} + \frac{\alpha^2 L^2 \lambda^2 \sigma^2}{\mu(1-\lambda)} + \frac{\alpha^4 L^4 \lambda^2 \sigma^2}{\mu(1-\lambda)^3 n}\right).$$

(35)

- For ATC-GT (6), we have

$$O\left(\frac{\alpha L \sigma^2}{\mu n} + \frac{\alpha^2 L^2 \lambda^2 \sigma^2}{\mu(1-\lambda)} + \frac{\alpha^4 L^4 \lambda^2 \sigma^2}{\mu(1-\lambda)^3 n}\right).$$

(36)

The bound (35) for ED/D$^2$ has the same network dependent bounds as in the strongly-convex case [14]. Moreover, the bound (36) for ATC-GT improves upon existing bounds for both strongly-convex [22] and PL settings [24], which are on the order of $O(\alpha^2 \lambda^2 \sigma^2/(1-\lambda)^3)$.

**VI. SIMULATION RESULTS**

In this section, we validate the established theoretical results with numerical simulations. Due to space limitations, the detailed experimental settings is given in the supplemental material where you can also find additional simulation results such as for PL condition case.

We consider the logistic regression problem with a non-convex regularization term [24], [61]. The problem formulation is given by

$$\min_{x \in \mathbb{R}^d} \frac{1}{n} \sum_{i=1}^{n} f_i(x) + \rho r(x),$$

where $f_i(x) = \frac{1}{L} \sum_{j=1}^{L} \ln(1 + e^{-y_i x_j})$ and $r(x) = \frac{\sum_{j=1}^{d} (x_j)^2}{2}$. In this problem, $x = \{x_{(j)}\}_{j=1}^{d} \in \mathbb{R}^d$ is the unknown variable to be optimized, $\{h_{i,\ell}, y_{i,\ell}\}_{i=1}^{L}$ is the training dataset held by agent $i$ in which $h_{i,\ell} \in \mathbb{R}^d$ is a feature vector while $y_{i,\ell} \in \{1,-1\}$ is the corresponding label. The regularization $r(x)$ is a smooth but non-convex function and the regularization constant $\rho > 0$ controls the influence of $r(x)$.

**Influence of network topology:** In this set of simulations, we will test the influence of the spectral gap $1-\lambda$ on various decentralized stochastic algorithms. We generate four topologies: the Erdos-Renyi graph with probability 0.8, the Ring topology, the Grid topology, and the scaled Ring topology with $\lambda = \frac{9 + \lambda_{\text{Ring}}}{10}$. The value of the mixing rate $\lambda$ for each topology is listed in the caption in Fig. 1. We utilize a constant step size 0.01 for each plot. It is observed in Fig. 1 that each decentralized algorithm will converge to a less accurate solution as $\lambda \to 1$ while PSgd is immune to the network topology. In addition, it is also observed that ED/D$^2$ is least sensitive to the network topology while DSGD is most sensitive (under heterogeneous setting), which is consistent with the results listed in Table I.
VII. CONCLUSION

In this work, we analyzed the convergence properties of SUDA (7) for decentralized stochastic non-convex optimization problems. SUDA is a general algorithmic framework that includes several state of the art decentralized methods as special case such as EXTRA, Exact-Diffusion/D², and gradient-tracking methods. We established the convergence of SUDA under both general non-convex and PL condition settings. Explicit convergence rate bounds are provided in terms of the problem parameters and the network topology. When specializing SUDA to the particular instances of ED/D², EXTRA, and GT-methods, we achieve improved network topology dependent rates compared to existing results under non-convex settings. Moreover, our rate shows that ED/D², EXTRA, and GT-methods are less sensitive to network topology compared to DSGD under heterogeneous data setting.

Finally, it should be noted that the lower bound from [23] suggests that it could be possible to further improve these network-dependent rates. However, such improvement have only been established by utilizing multiple gossip rounds as discussed in the introduction. Therefore, one potential future direction is to investigate whether these rates can be further improved without utilizing multiple gossip rounds.

APPENDIX A

RELATION OF (4) TO EXISTING METHODS

In this section, we describe how different existing methods are related to algorithm (4). First, note that we can equivalently describe the updates in (4) in terms of \( x \) by (see supp. material):

\[
\begin{align*}
\mathbf{x}_{k+2} &= \left( \mathbf{I} - \mathbf{B}^2 + \mathbf{AC} \right) \mathbf{x}_{k+1} - \mathbf{AC} \mathbf{x}_k - \alpha \mathbf{A} \left( \nabla f(\mathbf{x}_{k+1}) - \nabla f(\mathbf{x}_k) \right), \\
\end{align*}
\]

with \( \mathbf{x}^1 = \mathbf{A}(\mathbf{C}\mathbf{x}^0 - \alpha \nabla f(\mathbf{x}^0)) \). By choosing specific \( \mathbf{A}, \mathbf{B}, \mathbf{C} \) as a function of the combination matrix \( \mathbf{W} \), we can recover several important state-of-the-art methods. In the following, we do not assume that \( \mathbf{W} \) is symmetric unless otherwise stated.

Exact-Diffusion/D²: For Exact-Diffusion (ED) [12] (a.k.a. D² [15]), we assume \( \mathbf{W} \) to be symmetric and positive-semidefinite. If \( \mathbf{A} = \mathbf{W}, \mathbf{B} = (\mathbf{I} - \mathbf{W})^{1/2} \), and \( \mathbf{C} = \mathbf{I} \), we get ED/D² [12, 15]:

\[
\begin{align*}
\mathbf{x}_{k+2} &= \mathbf{W} \left( 2\mathbf{x}_{k+1} - \mathbf{x}_k - \alpha \left( \nabla f(\mathbf{x}_{k+1}) - \nabla f(\mathbf{x}_k) \right) \right).
\end{align*}
\]
\[(I - W)^{1/2}, \text{ and } C = W, \text{ we get EXTRA [11]:} \]
\[x^{k+2} = W \left(2x^{k+1} - x^k\right) - \alpha \left(\nabla f(x^{k+1}) - \nabla f(x^k)\right), \quad \text{(39)}\]

**ATC-GT:** Consider the adapt-then-combine GT method (ATC-GT) [16]:
\[x^{k+1} = W(x^k - \alpha g^k) \quad \text{(40a)}\]
\[g^{k+1} = W \left(g^{k} + \nabla f(x^{k+1}) - \nabla f(x^k)\right), \quad \text{(40b)}\]
with \(g^0 = W \nabla f(x^0)\) and \(x^0 = Wx^0\) (i.e., \(x^0\) is consensual). In a manner similar to [32], we can rewrite (40) equivalently as
\[x^{k+2} = 2Wx^{k+1} - W^2 x^k - \alpha W^2 \left(\nabla f(x^{k+1}) - \nabla f(x^k)\right), \quad \text{with } x^1 = W(x^0 - \alpha \nabla f(x^0)). \text{ The above is the same as (37) when } A = W^2, B^2 = (I - W)^2, \text{ and } C = I. \]

*Other GT variations:* Similar to [32], if we choose \(A = I, B^2 = (I - W)^2, \text{ and } C = W^2, \) then (37) becomes equivalent to non-ATC GT [18], [19]. Moreover, if we choose \(A = W, B^2 = (I - W)^2, \text{ and } C = W, \) then (37) becomes equivalent to the semi-ATC GT [17]. For interested readers, please see [58, Appendix A].

**APPENDIX B**

**Fundamental Factorization**

**A. Proof of Lemma 1**

Recall that \(\hat{\Lambda}_a = \text{diag}\{\lambda_{a,i}\}_{i=2}^n \otimes I_d, \quad \hat{\Lambda}_b = \text{diag}\{\lambda_{b,i}\}_{i=2}^n \otimes I_d, \quad \hat{\Lambda}_c = \text{diag}\{\lambda_{c,i}\}_{i=2}^n \otimes I_d. \) Hence, the matrix \(G\) defined in (21) can be rewritten as
\[G = \begin{bmatrix} \text{diag}\{\lambda_{a,i},\lambda_{c,i} - \lambda^2_{b,i}\}_{i=2}^n & -\text{diag}\{\lambda_{b,i}\}_{i=2}^n \\ \text{diag}\{\lambda_{b,i}\}_{i=2}^n & I_{n-1} \end{bmatrix} \otimes I_d. \]

Utilizing the structure of \(G, \) we can exchange the columns and rows of \(G\) through some permutation matrix \(P \in \mathbb{R}^{2d(n-1) \times 2d(n-1)}\) such that
\[\begin{bmatrix} \text{diag}\{\lambda_{a,i},\lambda_{c,i} - \lambda^2_{b,i}\}_{i=2}^n \\ \text{diag}\{\lambda_{b,i}\}_{i=2}^n \end{bmatrix} \otimes I_d. \]

\[\begin{bmatrix} 2\lambda_{a,i} - 1 - \sqrt{1 - \lambda_{i}^2} \\ \sqrt{1 - \lambda_{i}^2} \end{bmatrix} \in \mathbb{R}^{2 \times 2}. \quad \text{(41)}\]

Let us denote the eigenvalues of \(G_i\) by \(\gamma_{i,1}\) and \(\gamma_{i,2}. \) If the eigenvalues of \(G_{ij}\) are distinct \((\gamma_{i,1} \neq \gamma_{i,2}),\) then there exists a \(2 \times 2\) invertible \(V_i\) such that [62]:
\[G_i = V_i \Gamma_i V_i^{-1}, \quad \Gamma_i = \text{diag}\{\gamma_{i,1},\gamma_{i,2}\}. \]

It follows that \(G = \hat{V} \Gamma \hat{V}^{-1}\) where \(\hat{V} = P^T V, \quad V = \text{blkdiag}\{T_i E_i\}_{i=2}^n \otimes I_d, \quad \Gamma = \text{blkdiag}\{\Gamma_i\} \otimes I_d. \) Since the spectral radius of any matrix is upper bounded by any norm [62], it holds that
\[\|\Gamma_i\|_2^2 = \rho(\Gamma_i \Gamma_i^*) \leq \|\Gamma_i\|_1 \leq (|\gamma_i| + \epsilon)^2. \quad \text{(43)}\]

Here, \(\|\cdot\|_1\) denote the maximum-absolute-column-sum matrix norm. Therefore, \(\|\Gamma\| < 1\) for any \(\epsilon < 1 - |\gamma_i|. \)

**B2 Special Cases of Lemma 1**

In this section, we specify the results of Section B1 to the instances discussed in Appendix A.

**ED/D²:** We will consider the case for ED/D² (38) first. Recall from Appendix A that for ED/D², we have \(A = W, B = (I - W)^{1/2}, \) and \(C = I. \) For this case, we have \(\lambda_{a,i} = \lambda_i, \lambda_{c,i} = 1, \) and \(\lambda_{b,i} = \sqrt{1 - \lambda_i}\) where \(\lambda_i\) denote the eigenvalues of \(W. \) Thus, the matrix (41) becomes:
\[G_i = \begin{bmatrix} 2\lambda_{i} - 1 - \sqrt{1 - \lambda_{i}^2} \\ \sqrt{1 - \lambda_{i}^2} \end{bmatrix} \in \mathbb{R}^{2 \times 2}. \]

The eigenvalues of \(G_i\) \((i = 2, \ldots, n)\) are \(\gamma_{i,1,2} = \lambda_i \pm \sqrt{\lambda_i^2 - \lambda_i}. \) Note that \(\gamma_{i,1,2} < 1\) if \(-1 < \lambda_i < 1. \) This implies that for the convergence of ED/D², we require \(W > -1/2 I. \) It is sufficient for our purposes to assume that \(W \geq 0. \) Thus, we have two cases for the decomposition of \(G_i\) derived next.

- If \(0 < \lambda_i < 1, \) then the eigenvalues are complex and distinct:
\[\gamma_{i,1,2} = \lambda_i \pm j \sqrt{\lambda_i^2 - \lambda_i}, \quad |\gamma_{i,1,2}| = \sqrt{\lambda_i} < 1, \]
where \(j^2 = -1. \) We can factor \(G_i\) as \(G_i = V_i \text{diag}\{\gamma_{i,1,2}\} V_i^{-1}\) where
\[V_i = \begin{bmatrix} 1 & -1 \\ \sqrt{1 - \lambda_i} + j \sqrt{1 - \lambda_i} & \sqrt{1 - \lambda_i} - j \sqrt{1 - \lambda_i} \end{bmatrix}. \]

The inverse of \(V_i\) is
\[V_i^{-1} = \begin{bmatrix} 1 & -1 \\ \sqrt{1 - \lambda_i} - j \sqrt{1 - \lambda_i} & \sqrt{1 - \lambda_i} + j \sqrt{1 - \lambda_i} \end{bmatrix}. \]

The inverse of \(V_i\) is
\[V_i^{-1} = \begin{bmatrix} 1 & -1 \\ \sqrt{1 - \lambda_i} - j \sqrt{1 - \lambda_i} & \sqrt{1 - \lambda_i} + j \sqrt{1 - \lambda_i} \end{bmatrix}. \]

Note that
\[V_i V_i^* = \begin{bmatrix} 2 & -2 \sqrt{1 - \lambda_i} \\ 2 \sqrt{1 - \lambda_i} & 2 \end{bmatrix}. \]

Since the spectral radius of matrix is upper bounded by any of its norm, it holds that \(\|V_i\|^2 = \rho(V_i V_i^*) \leq \|V_i V_i^*\|_1 \leq 4.\) Following a similar argument for \(V_i^{-1}, \) we have \(\|V_i^{-1}\|^2 = \rho(V_i^{-1} V_i^{-1}) \leq \|V_i^{-1} V_i^{-1}\|_1 \leq 4.\)

- If \(\lambda_i = 0, \) then the eigenvalues of \(G_i\) are both zero \(\gamma_{i,1,2} = 0. \) Using Jordan canonical form, it holds that
\[T_i^{-1} G_i T_i = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}. \]

If we let \(E = \text{diag}\{1, \epsilon\} \) for \(0 < \epsilon < 1\) then
\[G_i = V_i \Gamma_i V_i^{-1}, \quad V_i = T_i E, \quad \Gamma_i = \begin{bmatrix} 1 & \frac{\epsilon}{2} \\ 0 & 0 \end{bmatrix}. \]
Simple calculations show that \( \| \Gamma_i \| = \epsilon < 1 \), \( \| V_i \|^2 \leq 2 \), and \( \| V_{i-1}^{-1} \|^2 \leq \frac{2}{\lambda} \). We can choose \( \epsilon^2 = \lambda = \max_{i \in \{2, \ldots, n\}} \lambda_i \).

Putting things together, we find

\[
\| \hat{V} \|^2 \leq 4, \quad \| \hat{V}^{-1} \|^2 \leq \frac{2}{\lambda} \quad \gamma = \| \Gamma \| = \sqrt{\lambda} \quad (44a)
\]

\[
\lambda_a = \| \hat{A}_a \| = \lambda, \quad \lambda_b = \frac{1}{\| \hat{A}_b \|} = \sqrt{1 - \lambda}, \quad (44b)
\]

where \( \lambda = \max_{i \in \{2, \ldots, n\}} \lambda_i \) and \( \lambda \) is the minimum non-zero eigenvalue of \( W \).

*EXTRA*: Observe that for EXTRA (39), the matrix \( G_i \) is identical to ED/D². Hence, the same bounds (44) hold for EXTRA except for \( \lambda_a \), which is equal to one, i.e., \( \lambda_a = 1 \), for EXTRA.

**ATC-GT**: Consider the ATC-GT method (6) (or 40). In this case, we have \( \lambda_{a,i} = \lambda_i^2, \lambda_{b,i} = 1 - \lambda_i, \) and \( \lambda_{c,i} = 1 \). Thus, the matrix \( G_i \) (\( i = 2, \ldots, n \)) given in (41) is

\[
G_i = \begin{bmatrix} 2\lambda_i - 1 & -1 - (1 - \lambda_i) \\ 1 - \lambda_i & 1 - \lambda_i \end{bmatrix} \in \mathbb{R}^{2 \times 2}.
\]

The eigenvalues of \( G_i \) are identical \( \gamma_{1,i} = \gamma_{2,i} = \lambda_i \). Clearly, the eigenvalues are strictly less than one by assumption since \( 0 \leq \lambda_i < 1 \) (\( i = 2, \ldots, n \)). If we let

\[
T_i = \begin{bmatrix} -1 & 0 \\ 1 & -\lambda_i \end{bmatrix}, \quad T_i^{-1} = \begin{bmatrix} -1 & 0 \\ 1 & 1 - \lambda_i \end{bmatrix},
\]

then, it holds that \( G_i = T_i \begin{bmatrix} \lambda_i & 0 \\ 0 & 1 \end{bmatrix} T_i^{-1}. \) If we further let \( E_i = \text{diag}\{1, \epsilon_i\} \) where \( \epsilon_i > 0 \) is an arbitrary constant, and define

\[
V_i = T_i E_i, \quad V_i^{-1} = \begin{bmatrix} -1 & 0 \\ 1 & 1 - \lambda_i \end{bmatrix}.
\]

Then, we have \( G_i = V_i \Gamma_i V_i^{-1} \) where

\[
\Gamma_i = \begin{bmatrix} \lambda_i & 1 \\ 0 & \lambda_i \end{bmatrix} E_i = \begin{bmatrix} \lambda_i & \epsilon \\ 0 & \lambda_i \end{bmatrix}.
\]

Choosing \( \epsilon = (1 - |\lambda_i|)/2 \), it can be verified that \( \| \Gamma \| < 1 \).

The above derivations are sufficient for our convergence analysis to cover GT methods with \( I < W \leq I \). However, we shall assume in the following that \( W \geq 0 \). Under this additional condition, we have \( \lambda_i \geq 0 \) and if we set \( \epsilon = (1 - \lambda_i)/2 \), it holds that

\[
\Gamma_i = \begin{bmatrix} \lambda_i & \frac{1-\lambda_i}{2} \\ 0 & \lambda_i \end{bmatrix}, \quad V_i = \begin{bmatrix} -1 & 0 \\ 1 & \frac{1}{2} \end{bmatrix}, \quad V_i^{-1} = \begin{bmatrix} -1 & 0 \\ 2 & 2 \end{bmatrix}.
\]

Hence, for ATC-GT with \( W \geq 0 \), we have

\[
\| \hat{V} \|^2 \leq 3, \quad \| \hat{V}^{-1} \|^2 \leq 9, \quad \gamma = \| \Gamma \| \leq \frac{1 + \lambda}{2} \quad (45a)
\]

\[
\lambda_a = \| \hat{A}_a \| = \lambda^2, \quad \lambda_b = \frac{1}{\| \hat{A}_b \|} = 1 - \lambda, \quad (45b)
\]

where \( \lambda = \max_{i \in \{2, \ldots, n\}} \lambda_i \).

**Other GT variants**: Note that for the other GT variations discussed in Appendix A, the matrix \( G_i \) is identical to the previous case. Hence, the same bounds (45) holds for these other variations except for the value of \( \lambda_a \), which is \( \lambda_a = 1 \) for the non-ATC-GT and \( \lambda_a = \lambda \) for the semi-ATC-GT variants.

**Appendix C**

**Convergence Proof**

In this section, we prove Theorems 1 and 2. We will first list some useful inequalities and facts that are used in our analysis.

**Useful inequalities**:

- Under the \( L \)-smoothness condition (10) given in Assumption 2, the aggregate cost \( f \) is also \( L \)-smooth. It follows that [63]:

\[
f(y) \leq f(z) + \langle \nabla f(z), y - z \rangle + \frac{L}{2} \| y - z \|^2, \quad (46)
\]

for all \( z, y \in \mathbb{R}^n \).

- Recall from (15) that \( w_i = \nabla f(x_i^k, \xi_i) - \nabla f(x_i^k) \) and \( w_i = \frac{1}{n} \sum_{j=1}^n (x_i^k, \xi_i) - \nabla f_i(x_i^k) \). Hence, under Assumption 3, we have

\[
E[\| w_i \|^2] = 0, \quad E[\| w_i \|^2] \leq n \sigma^2,
\]

\[
E[\| \hat{w} \|^2] = 0, \quad E[\| \hat{w} \|^2] \leq \frac{n \sigma^2}{n}.
\]

- Using (25), it holds that

\[
\| x_i^k - x_i^k \| = \| \hat{V} \| \| x_i^k \|^2 - \| \hat{A}_b^{-1} \hat{U} \|^2 \leq \| \hat{V} \|^2 \| x_i^k \|^2.
\]

Moreover, we have

\[
\| \nabla f(x_i^k) - \nabla f_i(x_i^k) \| \leq \frac{2}{n} \| x_i^k - x_i^k \| \leq \frac{2}{n} \| x_i^k - x_i^k \|^2 \leq \| \hat{V} \|^2 \| x_i^k \|^2.
\]

- Let \( v = \sqrt{n} v_2 \) where \( v_2 = \| \hat{V} \| \). Using the definition of \( \hat{e}^0 \) in (24), we have

\[
\| \hat{e}^0 \|^2 \leq \frac{\epsilon^2}{\epsilon^2} \left( \| \hat{U} \|^2 + \| \hat{A}_b^{-1} \hat{U} \|^2 \right)^2 = \frac{\lambda_b^2}{\lambda_b^2} \| \hat{U} \|^2 + \frac{\lambda_b^2}{\lambda_b^2} \| \hat{A}_b^{-1} \hat{U} \|^2.
\]

If the initialization is identical \( x_0^0 = 1 \times x_0^0 \) (for some \( x_0^0 \in \mathbb{R}^d \)), then \( \| \hat{U} \|^2 = 0 \) and \( x_0^0 = \hat{x}_0^0 \); moreover, \( x_0^0 = 0 \) since \( y_0 = 0 \). Hence, we have

\[
\| \hat{e}^0 \|^2 \leq \frac{\lambda_b^2}{\lambda_b^2} \| \hat{U} \|^2 + \frac{\lambda_b^2}{\lambda_b^2} \| \hat{A}_b^{-1} \hat{U} \|^2 = \frac{\lambda_b^2}{\lambda_b^2} \left( A - \frac{1}{n} I \right) \nabla f(x_0^0) \|^2 = \frac{\lambda_b^2}{\lambda_b^2} \left( A - \frac{1}{n} I \right) \nabla f(x_0^0) \|^2 \quad (50)
\]

where \( \zeta_0 = \frac{1}{n} \| \left( A - \frac{1}{n} I \right) \nabla f(x_0^0) - 1 \) \( \| \hat{e}^0 (x_0^0) \| \).

- For any \( 0 \leq \eta < 1 \) and non-negative sequence \( \{ w \} \), it holds that

\[
\sum_{\ell=0}^{k-1} \eta^{k-1-\ell} \leq \sum_{\ell=0}^{k-1} \eta^{k-1-\ell} \leq \frac{1}{1 - \eta}.
\]
\[
\sum_{k=1}^{K} \sum_{l=0}^{k-1} \eta^{k-l-1} u^{l} \leq \frac{1}{1-\eta} \sum_{k=0}^{K-1} w_{k}.
\]

(52)

A. Descent and Consensus Inequalities

In this section, we establish two inequalities given in Lemmas 3 and 4 that are essential to prove Theorems 1 and 2. We start with the following lemma regarding the iterates average.

**Lemma 3 (Descent inequality):** Suppose that Assumptions 2 and 3 holds. If \( \alpha \leq \frac{1}{\Gamma L} \), then for any \( k \geq 0 \), we have

\[
\begin{align*}
E[f(\tilde{x}^{k+1})] &\leq E[f(\tilde{x}^{k})] - \alpha \mathbb{E} \left[ \sum_{i=1}^{n} \nabla f_i(\tilde{x}^{k}) \right] + \frac{\alpha^2}{2} \mathbb{E} \left[ \sum_{i=1}^{n} \nabla^2 f_i(\tilde{x}^{k}) \right] \\
&\quad + \frac{\alpha^2 L^2}{2n} \mathbb{E} \left[ \sum_{i=1}^{n} \| \nabla^2 f_i(\tilde{x}^{k}) \|_{\infty}^2 \right] + \frac{\alpha^2}{2} \mathbb{E} \left[ \sum_{i=1}^{n} \| \nabla f_i(\tilde{x}^{k}) \|_{\infty}^2 \right].
\end{align*}
\]

(53)

where \( v_1 \triangleq \| \bar{V} \| \) and \( v \) is an arbitrary strictly positive constant.

**Proof:** Let

\[
\begin{align*}
\langle \bar{V}_i \rangle &\triangleq \frac{1}{\eta^k} \sum_{l=0}^{k-1} \hat{\bar{V}}_i^{l+1}, \\
\hat{\bar{V}}_i &\triangleq \frac{1}{\eta} \sum_{j=1}^{n} \frac{\nabla f_j(\bar{x}^{l+1})}{L} + \frac{\eta}{\eta^k} \bar{V}_i^{l+1} - \frac{\eta}{\eta^k} \bar{V}_i^{l+1}.
\end{align*}
\]

From (32),(33), we have

\[
\begin{align*}
\hat{\bar{V}}_i^{l+1} &\triangleq \frac{1}{\eta} \sum_{j=1}^{n} \frac{\nabla f_j(\bar{x}^{l+1})}{L} + \frac{\eta}{\eta^k} \bar{V}_i^{l+1} - \frac{\eta}{\eta^k} \bar{V}_i^{l+1}, \\
&\quad + \frac{\eta}{\eta^k} \bar{V}_i^{l+1}.
\end{align*}
\]

where \( \bar{V}_i^{l+1} \) is the left part of \( \bar{V}^{l+1} = [\bar{V}_1^{l+1}, \bar{V}_2^{l+1}, \ldots, \bar{V}_n^{l+1}] \). Using Cauchy-Schwarz inequality and \( ab \leq \frac{a^2}{2} + \frac{b^2}{2} \) for any non-negative scalars \( a \) and \( b \), the last term can be upper bounded by:

\[
\frac{2\alpha^2}{v^2} \mathbb{E} \left[ \sum_{i=1}^{n} \| \bar{V}_i^{l+1} \|_{\infty}^2 \right] + \frac{\alpha^2}{v^2} \mathbb{E} \left[ \sum_{i=1}^{n} \| \bar{V}_i^{l+1} \|_{\infty}^2 \right]
\]

Combining the last two equations and taking conditional expectation, we have

\[
\begin{align*}
E[f(\tilde{x}^{k+1})] &\leq f(\tilde{x}^{k}) - \alpha \mathbb{E} \left[ \sum_{i=1}^{n} \nabla f_i(\tilde{x}^{k}) \right] + \frac{\alpha^2}{2} \mathbb{E} \left[ \sum_{i=1}^{n} \nabla^2 f_i(\tilde{x}^{k}) \right] \\
&\quad + \frac{\alpha^2 L^2}{2n} \mathbb{E} \left[ \sum_{i=1}^{n} \| \nabla^2 f_i(\tilde{x}^{k}) \|_{\infty}^2 \right] + \frac{\alpha^2}{2} \mathbb{E} \left[ \sum_{i=1}^{n} \| \nabla f_i(\tilde{x}^{k}) \|_{\infty}^2 \right].
\end{align*}
\]

(54)

Substituting the bound (49) into inequality (54), letting \( \alpha \leq \frac{1}{\Gamma L} \), and taking the expectation yields (53).

We next establish an inequality regarding the consensus iterates \( \{\hat{\bar{V}}_i^{k}\} \).

**Lemma 4 (Consensus inequality):** Suppose that Assumptions 1–3 hold. Then, for all \( k \geq 0 \), we have

\[
\begin{align*}
E[\| \hat{\bar{V}}_i^{k+1} \|_{\infty}^2] &\leq \left( \frac{\gamma + \frac{2\alpha^2 L^2 \gamma^2 \lambda^2 \gamma}{1-\gamma}}{1-\gamma} \right) E[\| \hat{\bar{V}}_i^{k} \|_{\infty}^2] \\
&\quad + \frac{\alpha^2 L^2 \lambda^2 \gamma}{\lambda^2 (1-\gamma) v^2} E[\| \nabla f_i(\bar{x}^{k}) \|_{\infty}^2] \\
&\quad + \frac{\alpha^2 L^2 \lambda^2 \gamma}{\lambda^2 (1-\gamma) v^2} + \frac{\alpha^2 L^2 \lambda^2 \gamma}{\lambda^2 (1-\gamma) v^2}.
\end{align*}
\]

(55)

where \( v_1 \triangleq \| \bar{V} \| \), \( v_2 \triangleq \| \bar{V}^{-1} \| \), \( \lambda_n \triangleq \| \Lambda_n \| \), \( \gamma \triangleq \| \Gamma \| < 1 \), and \( \lambda_n \triangleq \frac{1}{1-\gamma} \).

**Proof:** Let

\[
\hat{\bar{V}}_i^{k+1} \triangleq \left[ \frac{\hat{\Lambda}_i^{k+1} \hat{\bar{U}}^T (\nabla f(\bar{x}^{k}) - \nabla f(\bar{x}^{k+1}))}{\Lambda_n^{k+1} \hat{\bar{U}}^T (\nabla f(\bar{x}^{k}) - \nabla f(\bar{x}^{k+1}))} \right]
\]

(56)

From (32),(33), we have

\[
\begin{align*}
\nabla f(\bar{x}^{k+1}) &\leq \nabla f(\bar{x}^{k}) - \alpha \bar{V}^{k+1} - \alpha \bar{V}^{k-1} \Lambda_n^{k+1} \bar{U}^T w_k \leq \nabla f(\bar{x}^{k}) - \alpha \bar{V}^{k+1} - \alpha \bar{V}^{k-1} \Lambda_n^{k+1} \bar{U}^T w_k.
\end{align*}
\]

(57)

where we used \( \| \bar{U} \| \leq 1 \) and defined \( \lambda_n^{k+1} \triangleq \| \Lambda_n^{k+1} \| \). We can bound the first term on the right hand side by using the inequality \( \| a + b \|_{\infty} \leq \| a \|_{\infty} + \| b \|_{\infty} \) for any \( \| a \|_{\infty} \leq 1 \). Doing so...
with $t = \gamma = \|\mathbf{f}\| < 1$ and using $1 \leq 1/(1-\gamma)$, we obtain
\[
E\|\hat{\mathbf{e}}^{k+1}\|^2 \leq \gamma E\|\hat{\mathbf{e}}^k\|^2 + \frac{2\gamma^2 \mathcal{L}_0^2 \mathcal{L}_{\alpha}^2}{(1-\gamma)v^2} (E\|\nabla \mathbf{f}(x^k) - \nabla \mathbf{f}(x^{k+1})\|^2
\]
\[+ \frac{1}{\lambda_\alpha^2} E\|\nabla \mathbf{f}(x^k) - \nabla \mathbf{f}(x^{k+1})\|^2 + \frac{2\alpha^2 \|\hat{\mathbf{V}}_{l}^{-1}\|^2 \lambda_\alpha^2 n \sigma^2}{v^2} \leq \gamma E\|\hat{\mathbf{e}}^k\|^2 + \frac{2\gamma^2 \mathcal{L}_0^2 \mathcal{L}_{\alpha}^2}{(1-\gamma)v^2} E\|x^k - x^{k+1}\|^2 + \frac{2\gamma^2 \mathcal{L}_0^2 \mathcal{L}_{\alpha}^2 n \sigma^2}{v^2} \leq \gamma E\|\hat{\mathbf{e}}^k\|^2 + \frac{2\gamma^2 \mathcal{L}_0^2 \mathcal{L}_{\alpha}^2 n \sigma^2}{v^2} \leq \bar{\gamma} E\|\hat{\mathbf{e}}^k\|^2, \tag{58}\]
where the second inequality holds due to smoothness condition (10) and $\|\hat{\mathbf{V}}_{l}^{-1}\| \leq v_2$. The second term can be bounded by using
\[
E\|[\bar{x}^k - \bar{x}^{k+1}]^2|\mathbf{f}^k\| \leq E\|[\alpha \nabla \mathbf{f}(x^k) + \alpha \hat{\mathbf{w}}^k]^2|\mathbf{f}^k| \leq \alpha^2 \|\nabla \mathbf{f}(x^k)\|^2 + \frac{\alpha^2 \sigma^2}{n}. \tag{57}\]
Taking the expectation and substituting the resulting bound into (58) yields inequality (55). \hfill \Box

**B. Proof of Theorem 1 (Non-Convex Case)**

**Proof of Theorem 1:** We start by deriving an ergodic bound on the consensus iterates $\{\hat{\mathbf{e}}^k\}$. Setting $\nu = \sqrt{n}v_2$ in (55), we get
\[
E\|\hat{\mathbf{e}}^{k+1}\|^2 \leq \left( \gamma + \frac{2\alpha^2 L^2 v^2 \mathcal{L}_0^2}{1-\gamma} \right) E\|\hat{\mathbf{e}}^k\|^2 + \frac{2\alpha^4 L^2 \mathcal{L}_0^2}{\lambda_\alpha^2 (1-\gamma)} E\|\nabla \mathbf{f}(x^k)\|^2 + \frac{2\alpha^2 \mathcal{L}_0^2 \mathcal{L}_{\alpha}^2 n \sigma^2}{v^2} \leq \gamma E\|\hat{\mathbf{e}}^k\|^2 + \frac{2\alpha^4 L^2 \mathcal{L}_0^2}{\lambda_\alpha^2 (1-\gamma)} \sum_{\ell=0}^{k-1} \frac{1+\gamma}{2} \left( 1 - \frac{\gamma}{2} \right)^{k-\ell} \nu^\ell \tag{59}\]
If the step size $\alpha$ satisfies
\[
\gamma + \frac{2\alpha^2 L^2 v^2 \mathcal{L}_0^2}{1-\gamma} \leq \frac{1+\gamma}{2} \Rightarrow \alpha \leq \frac{1-\gamma}{2Lv_1v_2\mathcal{L}_0}, \tag{60}\]
then the previous bound can be upper bounded by
\[
E\|\hat{\mathbf{e}}^{k+1}\|^2 \leq \frac{1+\gamma}{2} E\|\hat{\mathbf{e}}^k\|^2 + \frac{2\alpha^2 \mathcal{L}_0^2 \mathcal{L}_{\alpha}^2}{\lambda_\alpha^2 (1-\gamma)n} E\|\nabla \mathbf{f}(x^k)\|^2 + \frac{2\alpha^4 L^2 \mathcal{L}_0^2 \mathcal{L}_{\alpha}^2 n \sigma^2}{\lambda_\alpha^2 (1-\gamma)n} \leq \gamma E\|\hat{\mathbf{e}}^k\|^2 + \frac{2\alpha^4 L^2 \mathcal{L}_0^2 \mathcal{L}_{\alpha}^2}{\lambda_\alpha^2 (1-\gamma)n} \sum_{\ell=0}^{k-1} \left( 1 + \frac{\gamma}{2} \right)^{k-\ell} \nu^\ell \leq \frac{1+\gamma}{2} E\|\hat{\mathbf{e}}^k\|^2 + \frac{4\alpha^2 \mathcal{L}_0^2 \mathcal{L}_{\alpha}^2}{1-\gamma} + \frac{4\alpha^4 L^2 \mathcal{L}_0^2 \mathcal{L}_{\alpha}^2 n \sigma^2}{\lambda_\alpha^2 (1-\gamma)^2 n} \leq \frac{1+\gamma}{2} \left( \frac{1+\gamma}{2} \right)^{K-1} \sum_{k=0}^{K-1} \nu^k, \tag{61}\]
Let $w^k \triangleq E\|\nabla f(x^k)\|^2 + E\|\nabla \mathbf{f}(x^k)\|^2$. Recursively applying the previous inequality, it holds (for any $k = 1, 2, \ldots$) that
\[
E\|\hat{\mathbf{e}}^k\|^2 \leq \frac{1+\gamma}{2} \left( \frac{1+\gamma}{2} \right)^{k-1} E\|\hat{\mathbf{e}}^0\|^2 + \sum_{\ell=0}^{k-1} \left( 1 + \frac{\gamma}{2} \right)^{k-\ell} \left( \frac{2\alpha^2 \mathcal{L}_0^2 \mathcal{L}_{\alpha}^2}{\lambda_\alpha^2 (1-\gamma)n} \right)^\ell \tag{57}\]
where $\bar{f}(\bar{x}) \triangleq f(\bar{x}) - f^*$. Summing over $k = 0, \ldots, K-1$, dividing by $K \geq 1$, and using $\bar{f}(\bar{x}) \leq 0$, it holds that
\[
\frac{1}{K} \sum_{k=0}^{K-1} E\|\hat{\mathbf{e}}^k\|^2 \leq \frac{\alpha f(\bar{x})}{nK} + \frac{2\alpha L \mathcal{L}_0^2 \mathcal{L}_{\alpha}^2}{K} + \frac{2L^2 v_1^2 v_2^2}{K} \sum_{k=0}^{K-1} E\|\hat{\mathbf{e}}^k\|^2. \tag{66}\]
Substituting inequality (64) into (66) and rearranging, we obtain
\[
\left(1 - \frac{8\alpha^4 L^4v_0^2\sigma^2\lambda_k^2}{\lambda_k^2(1-\gamma)^2}\right) \frac{1}{K} \sum_{k=0}^{K-1} \mathbb{E}[\|\nabla f(\tilde{x}^k)\|^2 + \mathbb{E}[\nabla \nabla^T(\mathbf{x}^k)]^2]
\leq \frac{4\tilde{f}(\tilde{x}^0)}{\alpha K} + \frac{2\alpha L\sigma^2}{n} + \frac{6L^2v_0^2v_2^2\|e_0\|^2}{(1-\gamma)K}
\]
\[
+ \frac{8\alpha^2 L^2v_0^2v_2^2\sigma^2}{1-\gamma} + \frac{8\alpha^4 L^4v_0^2v_2^2\lambda_k^2\sigma^2}{\lambda_k^2(1-\gamma)^2n}.
\]
(67)

If we set
\[
\frac{1}{2} \leq 1 - \frac{8\alpha^4 L^4v_0^2v_2^2\lambda_k^2}{\lambda_k^2(1-\gamma)^2} \implies \alpha \leq \frac{\sqrt{\lambda_0(1-\gamma)}}{2L\sqrt{\lambda_1v_0\lambda_k}}
\]
and use (50) in inequality (67), we arrive at (26). The step size conditions in the theorem follows from \(\alpha \leq \frac{1}{2L}\) (from Lemma 3) and the conditions (60) and (68).

\[\Box\]

\section*{C. Proof of Theorem 2 (PL Condition Case)}

To prove Theorem 2, we need the following result.

\textbf{Lemma 5:} Under Assumption 2, the following holds:
\[
\|\nabla f(x)\|^2 \leq 2L(f(x) - f^*), \quad \forall x \in \mathbb{R}^d
\]
(69)

\[\frac{1}{n} \sum_{i=1}^{n} f(x_i) - f^* \leq \frac{L}{n} \|\nabla f(x)\|^2,
\]
(70)

\textbf{Proof:} Inequality (69) is proven in [64]. We include the proof here for convenience. Since \(f^* \leq f(z)\) for any \(z \in \mathbb{R}^d\), we have
\[
f^* \leq f\left(x - \frac{1}{L} \nabla f(x)\right) \leq f(x) - \frac{1}{2L} \|\nabla f(x)\|^2,
\]
(71)

where the last inequality holds due to (46) with \(y = x - \frac{1}{L} \nabla f(x)\) and \(z = x\). Rearranging the above we arrive at (69). We now establish inequality (70). The argument adjust the proof [24, Lemma 13] to our case. Substituting \(y = x_i\) and \(z = \tilde{x}\) in (46), we get
\[
f(x_i) \leq \tilde{f}(\tilde{x}) + \langle \nabla f(\tilde{x}), x_i - \tilde{x}\rangle + \frac{L}{2} \|x_i - \tilde{x}\|^2.
\]

Using Cauchy-Schwarz and Young’s inequalities, we have
\[
\langle \nabla f(\tilde{x}), x_i - \tilde{x}\rangle \leq \|\nabla f(\tilde{x})\| \|x_i - \tilde{x}\| \leq \frac{1}{2} \|\nabla f(\tilde{x})\|^2 + \frac{L}{2} \|x_i - \tilde{x}\|^2,
\]
(69)

Combining the last two bounds, then subtracting \(f^*\) from both sides and averaging over \(i\), we get
\[
\frac{1}{n} \sum_{i=1}^{n} (f(x_i) - f^*) \leq \frac{L}{2} \|x_i - \tilde{x}\|^2.
\]

Using (48), we arrive at (70). \[\Box\]

\textbf{Proof of Theorem 2:} Setting \(\nu = \sqrt{\lambda_1v_0}\) in (53) then subtracting \(f^*\) from both sides, using the PL inequality (12) and the bound \(-\frac{\mu}{2} \|\nabla \nabla^T(\mathbf{x}^k)\|^2 \leq 0\), we get
\[
\tilde{E}[\tilde{f}(\tilde{x}^{k+1})] \leq (1 - \mu \alpha) \tilde{E}[\tilde{f}(\tilde{x}^k)] + \frac{\alpha L^2v_0^2v_2^2}{2} \mathbb{E}[\|e_k\|^2] + \frac{\alpha^2 L^2\sigma^2}{2n}
\]
(72)

where \(\tilde{f}(\tilde{x}^k) = f(\tilde{x}^k) - f^*\). To establish our result, we will combine the above bound with a later bound given in (76), which we establish next. Note that
\[
\|\nabla \nabla^T(\mathbf{x}^k)\|^2 = \|\nabla \nabla^T(\mathbf{x}^k) - \nabla f(\tilde{x}^k) + \nabla f(\tilde{x}^k)\|^2
\]
\[
\leq 2\|\nabla \nabla^T(\mathbf{x}^k) - \nabla f(\tilde{x}^k)\|^2 + 2\|\nabla f(\tilde{x}^k)\|^2
\]
\[
\leq 2v_0^2v_2^2L^2\|e^k\|^2 + 4L\tilde{f}(\tilde{x}^k),
\]
where the last step holds due to (49) and (69). Substituting the above into (59), we get
\[
\mathbb{E}[\|e^k\|^2] \leq \frac{8\alpha^4 L^3\lambda_0^2}{\lambda_k^2(1-\gamma)} \tilde{E}[\tilde{f}(\tilde{x}^k)]
\]
\[
+ \left(\gamma + \frac{2\alpha^2 L^2v_0^2v_2^2\lambda_k^2}{1-\gamma} + \frac{4\alpha^4 L^4v_0^2v_2^2\lambda_k^2}{\lambda_k^2(1-\gamma)^2n}\right) \mathbb{E}[\|e^k\|^2]
\]
\[
+ 2\alpha^2\lambda_k^2\sigma^2 + 2\alpha^4 L^2\lambda_k^2\sigma^2.
\]
(73)

To simplify later expressions, we choose \(\alpha\) such that
\[
\frac{2\alpha^2 L^2v_0^2v_2^2\lambda_k^2}{1-\gamma} + \frac{4\alpha^4 L^4v_0^2v_2^2\lambda_k^2}{\lambda_k^2(1-\gamma)^2n} \leq \frac{3\alpha^2 L^2v_0^2v_2^2\lambda_k^2}{1-\gamma},
\]
(74a)

\[
\gamma + \frac{8\alpha^4 L^3\lambda_0^2}{\lambda_k^2(1-\gamma)^2n} \leq \frac{1 + \gamma}{2},
\]
(74b)

\[
\alpha \leq \min\left\{\frac{\lambda_0}{2L}, \frac{1-\gamma}{\sqrt{6L}v_0v_2\lambda_k}, \left(\frac{\mu\lambda^2_0(1-\gamma)}{8L^3v_0^2v_2^2}\right)^{1/3}\right\}.
\]
(75)

Under the previous conditions on \(\alpha\), inequality (73) is upper bounded by
\[
\mathbb{E}[\|e^{k+1}\|^2] \leq \frac{\alpha\mu^2}{L^2v_0^2} \tilde{E}[\tilde{f}(\tilde{x}^k)] + \left(\frac{1 + \gamma}{2}\right) \mathbb{E}[\|e^k\|^2]
\]
\[
+ 2\alpha^2\lambda_k^2\sigma^2 + 2\alpha^4 L^2\lambda_k^2\sigma^2.
\]
(76)

Note that from (70) with \(\nu = \sqrt{\lambda_1v_0}\), we have
\[
\frac{1}{n} \sum_{i=1}^{n} E[f(x_i) - f^*] \leq 2E[f(\tilde{x}^k) - f^*] + Lv_0^2v_2^2E[\|e_k\|].
\]
(77)

Multiplying inequality (72) by \(2\) and inequality (76) by \(Lv_0^2v_2^2\) and rewriting them in matrix form, we have
\[
\left[\begin{array}{c}
2E[\tilde{f}(\tilde{x}^{k+1})] \\
Lv_0^2v_2^2E[\|e^{k+1}\|^2]
\end{array}\right] \leq \left[\begin{array}{c}
\frac{1}{1+\gamma} \frac{\alpha\mu^2}{2} \\
\alpha L^2v_0^2v_2^2E[\|e^k\|^2]
\end{array}\right]
\]
\[\tilde{H}\]
\[
+ \left[\begin{array}{c}
\frac{\alpha^2 L^2\sigma^2}{n} \\
2\alpha^2 L^2v_0^2v_2^2\lambda_k^2\sigma^2 + 2\alpha^4 L^3v_0^2v_2^2\lambda_k^2\sigma^2
\end{array}\right].
\]
(78)
We will establish the convergence of \( \frac{1}{n} \sum_{i=1}^{n} E[f(x_i^k) - f^*] \) through the convergence of (78). Note that
\[
\rho(H) \leq \|H\|_1 = \max \left\{ 1 - \mu \alpha + \frac{\alpha \mu \lambda^2}{2}, \frac{1 + \gamma}{2} + \alpha L \right\} 
\leq 1 - \frac{\mu \alpha}{2},
\]
where the last inequality holds under the step size condition:
\[
\alpha \leq 1 - \frac{\gamma}{3L} \leq \frac{1 - \gamma}{2L + \mu}, \quad (80)
\]
Hence, \( \rho(H) < 1 \) and iterating inequality (78), we get
\[
\begin{align*}
&\frac{2}{n} \sum_{i=1}^{n} E[f(x_i^k) - f^*] \leq \|H\|_1 c_0 + \|(I - H)^{-1}h\|_1. \\
\end{align*}
\]
where \( c_0 = 2E\tilde{f}(\tilde{x}^0) + Lv^2_1 v^2_2 E[\tilde{e}^0] \leq 2E\tilde{f}(\tilde{x}^0) + \sum_{i=1}^{n} E[f(x_i^k) - f^*] \). We now bound the last term on the right. Following basic calculations, we can show that (see [58])
\[
\|(I - H)^{-1}h\|_1 \leq O \left( \frac{\alpha L \sigma^2}{\mu n} + \frac{\alpha^2 L_1 v^2_1 v^2_2 \lambda_0^2 \sigma^2}{\mu (1 - \gamma)} + \frac{\alpha^2 L_1 v^2_1 v^2_2 \lambda_0^2 \sigma^2}{\mu (1 - \gamma)^2 n} \right).
\]
Substituting the above into (82) and using (79), we obtain
\[
\frac{1}{n} \sum_{i=1}^{n} E[f(x_i^k)] \leq \left( 1 - \frac{\alpha \mu}{k} \right) c_0 + \frac{O \left( \frac{\alpha L \sigma^2}{\mu n} + \frac{\alpha^2 L_1 v^2_1 v^2_2 \lambda_0^2 \sigma^2}{\mu (1 - \gamma)} + \frac{\alpha^2 L_1 v^2_1 v^2_2 \lambda_0^2 \sigma^2}{\mu (1 - \gamma)^2 n} \right)}{\lambda_0^2}.
\]
Note that if \( x^0 = 1 \otimes x^0 \), then
\[
c_0 = 2E\tilde{f}(\tilde{x}^0) + Lv^2_1 v^2_2 E[\tilde{e}^0] \leq 2E\tilde{f}(\tilde{x}^0) + \frac{\alpha^2 L_1 v^2_1 v^2_2 \lambda_0^2 \sigma^2}{\lambda_0^2}.
\]
Combining the last two inequalities we get (30); moreover, combining the step size conditions (75) and (80), we get the step size condition in Theorem 2.

\[
\text{REFERENCES}
\]
\[1\] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein, "Distributed optimization and statistical learning via alternating direction method of multipliers," Found. Trends Mach. Learn., vol. 3, no. 1, pp. 1–122, Jan. 2011.
\[2\] M. Li et al., "Scaling distributed machine learning with the parameter server," in Proc. 11th USENIX Symp. Operating Syst. Des. Implementation, 2014, pp. 583–598.
\[3\] X. Lian, C. Zhang, H. Zhang, C.-J. Hsieh, W. Zhang, and J. Liu, "Can decentralized algorithms outperform centralized algorithms? A case study for decentralized parallel stochastic gradient descent," in Proc. Adv. Neural Inf. Process. Syst., 2017, pp. 5330–5340.
\[4\] M. Assran, N. Loizou, N. Ballas, and M. Rabbat, "Stochastic gradient push for distributed deep learning," in Proc. 36th Int. Conf. Mach. Learn., 2019, pp. 344–353.
\[5\] Y. Chen, K. Yuan, Y. Zhang, P. Pan, Y. Xu, and W. Yin, "Accelerating gossip SGD with periodic global averaging," in Proc. Int. Conf. Mach. Learn., 2021, pp. 1791–1802.
\[6\] S. S. Ram, A. Nedic, and V. V. Veeravalli, "Distributed stochastic subgradient projection algorithms for convex optimization," J. Optim. Theory Appl., vol. 147, no. 3, pp. 516–545, 2010.
\[7\] F. S. Cattivelli and A. H. Sayed, "Diffusion LMS strategies for distributed estimation," IEEE Trans. Signal Process., vol. 58, no. 3, pp. 1035–1048, Mar. 2010.
\[8\] J. Chen and A. H. Sayed, "Distributed pareto optimization via diffusion strategies," IEEE J. Sel. Topics Signal Process., vol. 7, no. 2, pp. 205–220, Apr. 2013.
\[9\] A. Nedic and A. Ozdaglar, "Distributed subgradient methods for multi-agent optimization," IEEE Trans. Autom. Control, vol. 54, no. 1, pp. 48–61, Jan. 2009.
\[10\] K. Yuan, Q. Ling, and W. Yin, "On the convergence of decentralized gradient descent," SIAM J. Optim., vol. 26, no. 3, pp. 1835–1854, 2016.
\[11\] W. Shi, Q. Ling, G. Wu, and W. Yin, "EXTRA: An exact first-order algorithm for decentralized consensus optimization," SIAM J. Optim., vol. 25, no. 2, pp. 944–966, 2015.
\[12\] K. Yuan, B. Ying, X. Zhao, and A. H. Sayed, "Exact diffusion for distributed optimization and learning-part i: Algorithm development," IEEE Trans. Signal Process., vol. 67, no. 3, pp. 708–723, Feb. 2019.
\[13\] Z. Li, W. Shi, and M. Yan, "A decentralized proximal-gradient method with network independent step-sizes and separated convergence rates," IEEE Trans. Signal Process., vol. 67, no. 17, pp. 4494–4506, Sep. 2019.
\[14\] K. Yuan, S. A. Alghunaim, B. Ying, and A. H. Sayed, "On the influence of bias-correction on distributed stochastic optimization," IEEE Trans. Signal Process., vol. 68, pp. 4352–4367, 2020.
\[15\] H. Tang, X. Lian, M. Yan, C. Zhang, and J. Liu, "D2: Decentralized training over decentralized data," in Proc. Int. Conf. Mach. Learn., Stockholm, Sweden, 2018, pp. 4848–4858.
\[16\] J. Xu, S. Zhu, Y. C. Soh, and L. Xie, "Augmented distributed gradient methods for multi-agent optimization under uncoordinated constant step-sizes," in Proc. IEEE 54th Conf. Decis. Control, 2015, pp. 2055–2060.
\[17\] P. D. Lorenzo and G. Scutari, "NEXT: In-network nonconvex optimization," IEEE Trans. Signal Inf. Process. Netw., vol. 2, no. 2, pp. 120–136, Jun. 2016.
\[18\] G. Qi and N. Li, "Harnessing smoothness to accelerate distributed optimization," IEEE Trans. Control Netw. Syst., vol. 5, no. 3, pp. 1245–1260, Sep. 2018.
\[19\] A. Nedic, A. Olshevsky, and W. Shi, "Achieving geometric convergence for distributed optimization over time-varying graphs," SIAM J. Optim., vol. 27, no. 4, pp. 2597–2633, 2017.
\[20\] K. Yuan and S. A. Alghunaim, "Removing data heterogeneity influence enhances network topology dependence of decentralized SGD," 2021, arXiv:2105.08023.
\[21\] K. Huang and S. Pu, "Improving the transient times for distributed stochastic gradient methods," 2021, arXiv:2105.04851.
\[22\] S. Pu and A. Nedić, "Distributed stochastic gradient tracking methods," Math. Program., vol. 187, no. 1, pp. 409–457, 2021.
\[23\] Y. Lu and C. De Sa, "Optimal complexity in decentralized training," in Proc. Int. Conf. Mach. Learn., 2021, pp. 7111–7123.
\[24\] R. Xin, R. A. Khan, and S. Kar, "An improved convergence analysis for decentralized online stochastic non-convex optimization," IEEE Trans. Signal Process., vol. 69, pp. 1842–1858, 2021.
\[25\] R. Xin, R. A. Khan, and S. Kar, "Fast decentralized nonconvex finite-sum optimization with recursive variance reduction," SIAM J. Optim., vol. 32, no. 1, pp. 1–28, 2022.
\[26\] B. T. Polyak, "Gradient methods for minimizing functionals," USSR Comput. Math. Math. Phys., vol. 3, no. 4, pp. 864–878, 1963.
\[27\] H. Karimi, J. Nutini, and M. Schmidt, "Linear convergence of gradient and proximal-gradient methods under the polyak-lojasiewicz condition," in Proc. Joint Eur. Conf. Mach. Learn. Knowl. Discov. Databases, Riva del Garda, Italy, 2016, pp. 795–811.
\[28\] G. Scutari and Y. Sun, "Distributed nonconvex constrained optimization over time-varying digraphs," Math. Program., vol. 176, no. 1–2, pp. 497–544, 2019.
\[29\] R. Scaman, F. BACH, S. Bubeck, Y. Lee, and L. Massoulie, "Optimal convergence rates for convex distributed optimization in networks," J. Mach. Learn. Res., vol. 20, pp. 1–31, 2019.
[30] S. A. Alghunaim, K. Yuan, and A. H. Sayed, “A linearly convergent proximal gradient algorithm for decentralized optimization,” in Proc. Adv. Neural Inf. Process. Syst., Vancouver, Canada, 2019, pp. 2844–2854.

[31] Y. Arjevani, J. Bruna, B. Can, M. Gurbuzbalaban, S. Jegelka, and H. Lin, “IDEAL: Inexact DCentralized accelerated augmented lagrangian method,” Proc. Adv. Neural Inf. Process. Syst., vol. 33, pp. 20684–20695, 2020.

[32] S. A. Alghunaim, E. K. Ryu, K. Yuan, and A. H. Sayed, “Decentralized proximal gradient algorithms with linear convergence rates,” IEEE Trans. Autom. Control, vol. 66, no. 6, pp. 2787–2794, Jun. 2021.

[33] J. Xu, Y. Tian, Y. Sun, and G. Scutari, “Distributed algorithms for composite optimization: Unified framework and convergence analysis,” IEEE Trans. Signal Process., vol. 69, pp. 3555–3570, 2021.

[34] A. Sundararajan, B. Van Scoy, and L. Lessard, “A canonical form for first-order distributed optimization algorithms,” in Proc. Amer. Control Conf., 2019, pp. 4075–4080.

[35] A. Sundararajan, B. Van Scoy, and L. Lessard, “Analysis and design of first-order distributed optimization algorithms over time-varying graphs,” IEEE Trans. Control Netw. Syst., vol. 7, no. 4, pp. 1597–1608, Dec. 2020.

[36] D. Jakovetic, “A unification and generalization of exact distributed first-order methods,” IEEE Trans. Signal Inf. Process. Netw., vol. 5, no. 1, pp. 31–46, Mar. 2019.

[37] P. Bianchi and J. Jakubowicz, “Convergence of a multi-agent projected stochastic gradient algorithm for non-convex optimization,” IEEE Trans. Autom. Control, vol. 58, no. 2, pp. 391–405, Feb. 2013.

[38] J. Chen and A. H. Sayed, “On the learning behavior of adaptive networks - Part I: Transient analysis,” IEEE Trans. Inf. Theory, vol. 61, no. 6, pp. 3487–3517, Jun. 2015.

[39] J. Chen and A. H. Sayed, “On the learning behavior of adaptive networks - Part II: Performance analysis,” IEEE Trans. Inf. Theory, vol. 61, no. 6, pp. 3518–3548, Jun. 2015.

[40] A. H. Sayed, “Adaptation, learning, and optimization over networks,” Found. Trends Mach. Learn., vol. 7, no. 4–5, pp. 311–801, 2014.

[41] T. Tatarenko and B. Touri, “Non-convex distributed optimization,” IEEE Trans. Autom. Control, vol. 62, no. 8, pp. 3744–3757, Aug. 2017.

[42] B. Swenson, R. Murray, S. Kar, and H. V. Poor, “Distributed stochastic gradient descent: Nonconvexity, nonsmoothness, and convergence to local minima,” 2020, arXiv:2003.02818.

[43] S. Pu, A. Olshevsky, and I. C. Paschalidis, “A sharp estimate on the transient time of distributed stochastic gradient descent,” IEEE Trans. Autom. Control, to be published, doi: 10.1109/TAC.2021.3126253.

[44] S. Vlaski and A. H. Sayed, “Distributed learning in non-convex environments–Part II: Polynomial escape from saddle-points,” IEEE Trans. Signal Process., vol. 69, pp. 1257–1270, 2021.

[45] Z. Jiang, A. Balu, C. Hegde, and S. Sarkar, “Collaborative deep learning in fixed topology networks,” in Proc. Adv. Neural Inf. Process. Syst., Long Beach, CA, USA, 2017, pp. 5904–5914.

[46] A. Koloskova, S. Stich, and M. Jaggi, “Decentralized stochastic optimization and gossip algorithms with compressed communication,” in Proc. 39th Int. Conf. Mach. Learn., Long Beach, California, USA, 2019, pp. 3478–3487.

[47] A. Koloskova, N. Loizou, S. Boreiri, M. Jaggi, and S. Stich, “A unified theory of decentralized SGD with changing topology and local updates,” in Proc. Int. Conf. Mach. Learn., 2020, pp. 5381–5393.

[48] J. Wang and G. Joshi, “Cooperative SGD: A unified framework for the design and analysis of local-update SGD algorithms,” J. Mach. Learn. Res., vol. 22, no. 213, pp. 1–50, 2021.

[49] J. Zhang and K. You, “Decentralized stochastic gradient tracking for non-convex empirical risk minimization,” 2019, arXiv:1909.02712.

[50] S. Lu, X. Zhang, H. Sun, and M. Hong, “GNSD: A gradient-tracking based nonconvex stochastic algorithm for decentralized optimization,” in Proc. IEEE Data Sci. Workshop, 2019, pp. 315–321.

[51] S. Lu and C. W. Wu, “Decentralized stochastic non-convex optimization over weakly connected time-varying digraphs,” in Proc. IEEE Int. Conf. Acoust. Speech, Signal Process., 2020, pp. 5770–5774.

[52] X. Yi, S. Zhang, T. Yang, T. Chaif, and K. H. Johansson, “A primal-dual SGD algorithm for distributed nonconvex optimization,” IEEE/CAA J. Automática Sinica, vol. 9, no. 5, pp. 812–833, May 2022.

[53] R. Xin, S. Das, U. A. Khan, and S. Kar, “A stochastic proximal gradient framework for decentralized non-convex composite optimization: Topology-independent sample complexity and communication efficiency,” 2021, arXiv:2110.01594.

[54] P. Patrascuk and X. Yuan, “Bandwidth optimal all-reduce algorithms for clusters of workstations,” J. Parallel Distrib. Comput., vol. 69, no. 2, pp. 117–124, 2009.

[55] Y. You, I. Gitman, and B. Ginsburg, “Large batch training of convolutional networks,” 2017, arXiv:1706.05888.

[56] M. Gurbuzbalaban, U. Simsekli, and L. Zhu, “The heavy-tail phenomenon in SGD,” in Proc. Int. Conf. Mach. Learn., 2021, pp. 3964–3975.

[57] Q. Ling, W. Shi, G. Wu, and A. Ribeiro, “DLM: Decentralized linearized alternating direction method of multipliers,” IEEE Trans. Signal Process., vol. 63, no. 15, pp. 4051–4064, Aug. 2015.

[58] S. A. Alghunaim and K. Yuan, “A unified and refined convergence analysis for non-convex decentralized learning,” 2021, arXiv:2110.09993.

[59] Y. Tang, J. Zhang, and N. Li, “Distributed zero-order algorithms for nonconvex multiagent optimization,” IEEE Trans. Control Netw. Syst., vol. 8, no. 1, pp. 269–281, Mar. 2021.

[60] R. Xin, U. Khan, and S. Kar, “A hybrid variance-reduced method for decentralized stochastic non-convex optimization,” in Proc. 38th Int. Conf. Mach. Learn., 2021, pp. 11459–11469.

[61] A. Antoniadis, I. Gijsels, and M. Nikolova, “Penalized likelihood regression for generalized linear models with non-quadratic penalties,” Ann. Inst. Stat. Math., vol. 63, no. 3, pp. 585–615, 2011.

[62] R. A. Horn and C. R. Johnson, Matrix Analysis. Cambridge, U.K.: Cambridge Univ. Press, 2012.

[63] Y. Nesterov, Introductory Lectures on Convex Optimization: A Basic Course, vol. 87. Berlin, Germany: Springer, 2013.

[64] B. Polyak, Introduction to Optimization. New York, NY, USA: Optimization Software, Publications Division, 1987.

Sulaiman A. Alghunaim received the B.S. degree in electrical engineering from Kuwait University, Kuwait City, Kuwait, in 2013, the M.S. degree in electrical engineering, and the Ph.D. degree in electrical and computer engineering from the University of California, Los Angeles, Los Angeles, CA, USA, in 2016 and 2020, respectively. He is currently an Assistant Professor with Electrical Engineering Department, Kuwait University. His research interests include optimization and learning theory and their applications in multi-agent systems, signal processing, and control. His current research interests include development and analysis of distributed optimization and learning algorithms.

Kun Yuan (Member, IEEE) received the Ph.D. degree from the Department of Electrical and Computer Engineering, University of California, Los Angeles, Los Angeles, CA, USA, in 2019. Then he joined Alibaba (U.S.) Group as a Research Scientist. His research interests mainly include theory, algorithms, and applications in optimization, signal processing, and machine learning. Dr. Yuan was the recipient of the 2017 IEEE Signal Processing Society Young Author Best Paper Award, and the 2017 International Consortium of Chinese Mathematicians (ICCM) Distinguished Paper Award.