A Linearly Convergent Proximal Gradient Algorithm for Decentralized Optimization

Sulaiman A. Alghunaim*, Kun Yuan*, and Ali H. Sayed†

*Department of Electrical and Computer Engineering, University of California, Los Angeles
†School of Engineering, Ecole Polytechnique Federale de Lausanne, Switzerland

Introduction

Decentralized optimization is a powerful paradigm that finds applications in engineering and learning design. This work studies decentralized composite optimization problems with non-smooth regularization terms. Most existing gradient-based proximal decentralized methods are known to converge to the optimal solution with sublinear rates, and it remains unclear whether this family of methods can achieve global linear convergence. To tackle this problem, this work assumes the non-smooth regularization term is common across all networked agents, which is the case for many machine learning problems. Under this condition, we design a proximal gradient decentralized algorithm whose fixed point coincides with the desired minimizer. We then provide a concise proof that establishes its linear convergence. In the absence of the non-smooth term, our analysis technique covers the well known EXTRA algorithm and provides useful bounds on the convergence rate and step-size.

Problem Formulation

Consider a network of K agents (e.g., machines, processors) connected over some graph. Through only local interactions (i.e., agents only communicate with their immediate neighbors), each node is interested in finding a consensual vector, denoted by $w^\star$, that minimizes the following aggregate cost:

$$w^\star = \arg\min_{w\in\mathbb{R}^M} \quad \frac{1}{K}\sum_{k=1}^{K} J_k(w) + R(w) \tag{1}$$

The cost function $J_k(w) : \mathbb{R}^M \to \mathbb{R}$ is privately known by agent k and $R(w) : \mathbb{R}^M \to \mathbb{R}\cup\{\infty\}$ is a proper[a] and lower-semicontinuous convex function (not necessarily differentiable). We adopt the following assumption throughout this work.

***Assumption* 1.** ***(Cost function)**: There exists a solution $w^\star$ to problem (1). Moreover, each cost function $J_k(w)$ is convex and first-order differentiable with δ-Lipschitz continuous gradients:*

$$\|\nabla J_k(w^o) - \nabla J_k(w^\bullet)\| \le \delta\|w^o - w^\bullet\|, \quad \textit{for any } w^o \textit{ and } w^\bullet \tag{2}$$

and the aggregate cost function $\bar{J}(w) = \frac{1}{K}\sum_{k=1}^{K} J_k(w)$ is ν-strongly-convex:

$$(w^o - w^\bullet)^\mathsf{T}\Big(\nabla\bar{J}(w^o) - \nabla\bar{J}(w^\bullet)\Big) \ge \nu\|w^o - w^\bullet\|^2 \tag{3}$$

for any w^o and $w^\bullet$. The constants ν and δ satisfy $0 < \nu \le \delta$. □

Note that from the strong-convexity condition (3), we know the objective function in (1) is also strongly convex and, thus, the global solution $w^\star$ is unique.

[a] The function $f(.)$ is proper if $-\infty < f(x)$ for all x in its domain and $f(x) < \infty$ for at least one x.

Contribution

This paper considers the composite optimization problem (1) and has two main contributions. First, for the case of a common non-smooth regularizer $R(w)$ across all computing agents, we propose a proximal decentralized algorithm whose fixed point coincides with the desired global solution $w^\star$. We then provide a short proof to establish its linear convergence when the aggregate of the smooth functions $\sum_{k=1}^{K} J_k(w)$ is strongly convex. This result closes the existing gap between decentralized proximal gradient methods and the centralized proximal gradient methods. The second contribution is in our convergence proof technique. Specifically, we provide a concise proof that is applicable to general decentralized primal-dual gradient methods such as EXTRA [5] when $R(w) = 0$. Our proof provides useful bounds on the convergence rate and step-sizes.

Algorithm Derivation

We start by introducing the network weights that are used to implement the algorithm in a decentralized manner. Thus, we let a_{sk} denote the weight used by agent k to scale information arriving from agent s with $a_{sk} = 0$ if s is not a direct neighbor of agent k, i.e., there is no edge connecting them. Let $A = [a_{sk}] \in \mathbb{R}^{K\times K}$ denote the weight matrix associated with the network. Then, we assume A to be symmetric and doubly stochastic, i.e., $A\mathbb{1}_K = \mathbb{1}_K$ and $\mathbb{1}_K^\mathsf{T} A = \mathbb{1}_K^\mathsf{T}$. We also assume that A is primitive, i.e., there exists an integer p such that all entries of A^p are positive. Note that as long as the network is connected, there exist many ways to generate such weight matrices in a decentralized fashion – [1, 3, 6]. Under these conditions, it holds from the Perron-Frobenius theorem [2] that A has a single eigenvalue at one with all other eigenvalues being strictly less than one. Therefore, $(I_K - A)x = 0$ if, and only, if $x = c\mathbb{1}_K$ for any $c \in \mathbb{R}$. If we let $w_k \in \mathbb{R}^M$ denote a local copy of the global variable w available at agent k and introduce the network quantities:

$$\mathscr{w} \;\triangleq\; \mathrm{col}\{w_1, \cdots, w_K\} \in \mathbb{R}^{KM}, \quad \mathcal{B} \;\triangleq\; \frac{1}{2}(I_{KM} - A\otimes I_M) \tag{4}$$

then, it holds that $\mathcal{B}\mathscr{w} = 0$ if, and only if, $w_k = w_s$for all k, s. Note that since A is symmetric with eigen values between $(-1, 1]$, the matrix $\mathcal{B}$ is positive semi-definite with eigenvalues in $[0, 1)$. Problem (1) is equivalent to the following constrained problem:

$$\underset{\mathscr{w}\in\mathbb{R}^{KM}}{\text{minimize}} \quad \mathcal{J}(\mathscr{w}) + \mathcal{R}(\mathscr{w}), \quad \text{s.t. } \mathcal{B}^{\frac{1}{2}}\mathscr{w} = 0 \tag{5}$$

where $\mathcal{J}(\mathscr{w}) \;\triangleq\; \sum_{k=1}^{K} J_k(w_k)$, $\mathcal{R}(\mathscr{w}) \;\triangleq\; \sum_{k=1}^{K} R(w_k)$ and $\mathcal{B}^{\frac{1}{2}}$ is the square root of the positive semi-definite matrix $\mathcal{B}$. To solve problem (5), we introduce first the following equivalent saddle-point problem:

$$\min_{\mathscr{w}}\max_{\mathscr{y}} \quad \mathcal{L}_\mu(\mathscr{w}, \mathscr{y}) \;\triangleq\; \mathcal{J}(\mathscr{w}) + \mathcal{R}(\mathscr{w}) + \mathscr{y}^\mathsf{T}\mathcal{B}^{\frac{1}{2}}\mathscr{w} + \frac{1}{2\mu}\|\mathcal{B}^{\frac{1}{2}}\mathscr{w}\|^2 \tag{6}$$

where $\mathscr{y} \in \mathbb{R}^{MK}$ is the dual variable and $\mu > 0$ is the coefficient for the augmented Lagrangian. By introducing $\mathcal{J}_\mu(\mathscr{w}) = \mathcal{J}(\mathscr{w}) + 1/2\mu\|\mathcal{B}^{\frac{1}{2}}\mathscr{w}\|^2$, it holds that

$$\mathcal{L}_\mu(\mathscr{w}, \mathscr{y}) = \mathcal{J}_\mu(\mathscr{w}) + \mathcal{R}(\mathscr{w}) + \mathscr{y}^\mathsf{T}\mathcal{B}^{\frac{1}{2}}\mathscr{w}. \tag{7}$$

To solve the saddle point problem in (6), we propose the following recursion. For $i \ge 0$:

$$\begin{cases} \mathscr{z}_i = \mathscr{w}_{i-1} - \mu\nabla\mathcal{J}_\mu(\mathscr{w}_{i-1}) - \mathcal{B}^{\frac{1}{2}}\mathscr{y}_{i-1} & \text{(8a)}\\ \mathscr{y}_i = \mathscr{y}_{i-1} + \alpha\mathcal{B}^{\frac{1}{2}}\mathscr{z}_i & \text{(8b)}\\ \mathscr{w}_i = \mathbf{prox}_{\mu\mathcal{R}}(\mathscr{z}_i) & \text{(8c)} \end{cases}$$

where $\alpha > 0$ is the dual step-size (a tunable parameter). We will next show that with the initialization $\mathscr{y}_0 = 0$, we can implement this algorithm in a decentralized manner.

The Decentralized Implementation

From the definition of $\mathcal{J}_\mu(\mathscr{w})$, we have $\nabla\mathcal{J}_\mu(\mathscr{w}) = \nabla\mathcal{J}(\mathscr{w}) + 1/\mu\ \mathcal{B}\mathscr{w}$. Substituting $\nabla\mathcal{J}_\mu(\mathscr{w})$ into (8a), we have

$$\mathscr{z}_i = (I_{KM} - \mathcal{B})\mathscr{w}_{i-1} - \mu\nabla\mathcal{J}(\mathscr{w}_{i-1}) - \mathcal{B}^{\frac{1}{2}}\mathscr{y}_{i-1}, \tag{9}$$

With the above relation, we have for $i \ge 1$

$$\mathscr{z}_i - \mathscr{z}_{i-1} = (I - \mathcal{B})(\mathscr{w}_{i-1} - \mathscr{w}_{i-2}) - \mu\Big(\nabla\mathcal{J}(\mathscr{w}_{i-1}) - \nabla\mathcal{J}(\mathscr{w}_{i-2})\Big) - \mathcal{B}^{\frac{1}{2}}(\mathscr{y}_{i-1} - \mathscr{y}_{i-2}) \tag{10}$$

From (8b) we have $\mathscr{y}_{i-1} - \mathscr{y}_{i-2} = \alpha\mathcal{B}^{\frac{1}{2}}\mathscr{z}_{i-1}$. Substituting this relation into (10), we reach

$$\mathscr{z}_i = (I - \alpha\mathcal{B})\mathscr{z}_{i-1} + (I - \mathcal{B})(\mathscr{w}_{i-1} - \mathscr{w}_{i-2}) - \mu\Big(\nabla\mathcal{J}(\mathscr{w}_{i-1}) - \nabla\mathcal{J}(\mathscr{w}_{i-2})\Big) \tag{11}$$

for $i \ge 1$. For initialization, we can repeat a similar argument to show that the proximal primal-dual method (8a)–(8c) with $\mathscr{y}_0 = 0$ is equivalent to the following algorithm. Let $\mathscr{z}_0 = \mathscr{w}_{-1} = 0$, set $\nabla\mathcal{J}(\mathscr{w}_{-1}) \leftarrow 0$, and $\mathscr{w}_0$ to any arbitrary value. Repeat for $i = 1, \cdots$

$$\begin{aligned} \mathscr{z}_i &= (I - \alpha\mathcal{B})\mathscr{z}_{i-1} + (I - \mathcal{B})(\mathscr{w}_{i-1} - \mathscr{w}_{i-2}) - \mu\Big(\nabla\mathcal{J}(\mathscr{w}_{i-1}) - \nabla\mathcal{J}(\mathscr{w}_{i-2})\Big) && \text{(12a)}\\ \mathscr{w}_i &= \mathbf{prox}_{\mu\mathcal{R}}(\mathscr{z}_i) && \text{(12b)} \end{aligned}$$

Since $\mathcal{B}$ has network structure, recursion (12) can be implemented in a decentralized way. This algorithm only requires each agent to share one vector at each iteration; a per agent implementation of resulting proximal primal-dual diffusion (P2D2) algorithm is listed in (13).

Proximal Primal-Dual Diffusion (P2D2)

Let $B = 0.5(I - A) = [b_{sk}]$ and choose step-sizes μ and α. Set all initial variables to zero and repeat for $i = 1, 2, \cdots$

$$\begin{aligned} \phi_{k,i} &= \sum_{s\in\mathcal{N}_k} b_{sk}(\alpha z_{s,i-1} + w_{s,i-1} - w_{s,i-2}) \qquad \textbf{(Communication Step)} && \text{(13a)}\\ \psi_{k,i} &= w_{k,i-1} - \mu\nabla J_k(w_{k,i-1}) && \text{(13b)}\\ z_{k,i} &= z_{k,i-1} + \psi_{k,i} - \psi_{k,i-1} - \phi_{k,i} && \text{(13c)}\\ w_{k,i} &= \mathbf{prox}_{\mu R}(z_{k,i}) && \text{(13d)} \end{aligned}$$

Auxilary Results

We start by showing the existence and properties of a fixed point for recursions (8a)–(8c).

***Lemma* 1** (Fixed point optimilaty)**.** *Under Assumption 1, a fixed point $(\mathscr{w}^\star, \mathscr{y}^\star, \mathscr{z}^\star)$ exists for recursions (8a)–(8c), i.e., it holds that*

$$\begin{cases} \mathscr{z}^\star = \mathscr{w}^\star - \mu\nabla\mathcal{J}_\mu(\mathscr{w}^\star) - \mathcal{B}^{\frac{1}{2}}\mathscr{y}^\star & \textit{(14a)}\\ 0 = \mathcal{B}^{\frac{1}{2}}\mathscr{z}^\star & \textit{(14b)}\\ \mathscr{w}^\star = \mathbf{prox}_{\mu\mathcal{R}}(\mathscr{z}^\star) & \textit{(14c)} \end{cases}$$

Moreover, $\mathscr{w}^\star$ and $\mathscr{z}^\star$ are unique and each block element of $\mathscr{w}^\star = \mathrm{col}\{w_1^\star, \cdots, w_K^\star\}$ coincides with the unique solution $w^\star$ to problem (1), i.e., $w_k^\star = w^\star$ for all k.

From Lemma 1, we see that although $\mathscr{w}^\star$ and $\mathscr{z}^\star$ are unique, there can be multiple fixed points. This is because from (14a), $\mathscr{y}^\star$ is not unique due the rank deficiency of $\mathcal{B}^{\frac{1}{2}}$. However, by following similar arguments to the ones from [4], it can be verified that there exists a particular fixed point $(\mathscr{w}^\star, \mathscr{y}_b^\star, \mathscr{z}^\star)$ satisfying (14a)–(14c) where $\mathscr{y}_b^\star$ is a unique vector that belongs to the range space of $\mathcal{B}^{\frac{1}{2}}$. In the following we will show that the iterates $(\mathscr{w}_i, \mathscr{y}_i, \mathscr{z}_i)$ converge linearly to this particular fixed point $(\mathscr{w}^\star, \mathscr{y}_b^\star, \mathscr{z}^\star)$.
To establish the linear convergence of the proximal primal-dual diffusion (P2D2) (8a)–(8c) we introduce the error quantities:

$$\widetilde{\mathscr{w}}_i \;\triangleq\; \mathscr{w}_i - \mathscr{w}^\star, \quad \widetilde{\mathscr{y}}_i \;\triangleq\; \mathscr{y}_i - \mathscr{y}_b^\star, \quad \widetilde{\mathscr{z}}_i = \mathscr{z}_i - \mathscr{z}^\star \tag{15}$$

By subtracting (14a)–(14c) from (8a)–(8c) with $\mathscr{y}^\star = \mathscr{y}_b^\star$, we reach the following error recursions

$$\begin{cases} \widetilde{\mathscr{z}}_i = \widetilde{\mathscr{w}}_{i-1} - \mu\Big(\nabla\mathcal{J}_\mu(\mathscr{w}_{i-1}) - \nabla\mathcal{J}_\mu(\mathscr{w}^\star)\Big) - \mathcal{B}^{\frac{1}{2}}\widetilde{\mathscr{y}}_{i-1} & \text{(16a)}\\ \widetilde{\mathscr{y}}_i = \widetilde{\mathscr{y}}_{i-1} + \alpha\mathcal{B}^{\frac{1}{2}}\widetilde{\mathscr{z}}_i & \text{(16b)}\\ \widetilde{\mathscr{w}}_i = \mathbf{prox}_{\mu\mathcal{R}}(\mathscr{z}_i) - \mathbf{prox}_{\mu\mathcal{R}}(\mathscr{z}^\star) & \text{(16c)} \end{cases}$$

We let $\sigma_{\max}$ and $\underline{\sigma}$ denote the maximum singular value and minimum non-zero singular value of the matrix $\mathcal{B}$. Notice that from (4), $\mathcal{B}$ is symmetric and, thus, its singular values are equal to its eigenvalues and are in $[0, 1)$ (i.e., $\sigma_{\min} = 0 < \underline{\sigma} \le \sigma_{\max} < 1$). The following result follows from [5, Proposition 3.6].

***Lemma* 2** (Augmented Cost)**.** *Under Assumption 1, the penalized augmented cost $\mathcal{J}(\mathscr{w}) + \frac{\rho}{2}\|\mathscr{w}\|_{\mathcal{B}}^2$ with any $\rho > 0$ is restricted strongly-convex with respect to $\mathscr{w}^\star$:*

$$(\mathscr{w} - \mathscr{w}^\star)^\mathsf{T}\Big(\nabla\mathcal{J}(\mathscr{w}) - \nabla\mathcal{J}(\mathscr{w}^\star)\Big) + \rho\|\mathscr{w} - \mathscr{w}^\star\|_{\mathcal{B}}^2 \ge \nu_\rho\|\mathscr{w} - \mathscr{w}^\star\|^2 \tag{17}$$

where

$$\nu_\rho = \min\left\{\nu - 2\delta c, \frac{\rho\underline{\sigma}(\mathcal{B})c^2}{4(c^2+1)}\right\} > 0, \quad \textit{for any } c \in \left(0, \frac{\nu}{2\delta}\right) \tag{18}$$

for any $\mathscr{w}$ with $\mathscr{w}^\star = \mathbb{1}\otimes w^\star$ and where $w^\star$ denotes the minimizer of (1).

Main Result

***Theorem* 1** (Linear convergence)**.** *Under Assumption 1, $\mathscr{y}_0 = 0$, and if step-sizes satisfy*

$$\mu < \frac{(1 - \sigma_{\max})}{\delta}, \quad \alpha \le \min\{1, \mu\nu_\rho(2 - \sigma_{\max} - \mu\delta)\}, \tag{19}$$

It holds that $\|\widetilde{\mathscr{w}}_i\|^2 \le C\gamma^i$ where $C > 0$ and

$$\gamma \;\triangleq\; \max\{1 - \mu\nu_\rho(2 - \sigma_{\max} - \mu\delta)/(1 - \alpha\sigma_{\max}), 1 - \alpha\underline{\sigma}\} < 1. \tag{20}$$

for some $\rho > 0$ with ν_ρ given in (18).

Next we show that when $R(w) = 0$, we can have a better upper bound for the dual step-size, which covers the EXTRA algorithm [5].

***Theorem* 2** (Linear convergence when $R(w) = 0$)**.** *Under Assumption 1, if $R(w) = 0$, $\mathscr{y}_0 = 0$, and the step-sizes satisfy $\mu < \frac{(1-\sigma_{\max})}{\delta}$ and $\alpha \le 1$, it holds that $\|\widetilde{\mathscr{w}}_i\|_{\mathcal{Q}}^2 \le C\gamma^i$ where $C > 0$, $\mathcal{Q} = I - \alpha\mathcal{B} > 0$, and*

$$\gamma = \max\Big\{1 - \mu\nu_\rho(2 - \sigma_{\max} - \mu\delta), 1 - \alpha\underline{\sigma}\Big\} < 1$$

for some $\rho > 0$ with ν_ρ given in (18).

In the above Theorem, we see that the convergence rate bound is upper bounded by two terms, one term is from the cost function and the other is from the network. This bound shows how the network affects the convergence rate of the algorithm. For example, in Theorem 2, assume that $\alpha = 1$ and the network term dominates the convergence rate so that $\gamma = 1 - \alpha\underline{\sigma} = 1 - \underline{\sigma}$. Recall that $\underline{\sigma} = \underline{\sigma}(\mathcal{B})$ is the smallest non-zero singular value (or eigenvalue) of the matrix $0.5(I - A)$. Thus, the effect of the network on the convergence rate is evident through the term $1 - \underline{\sigma}$, which becomes close to one as the network becomes more sparse. Note when $\alpha = 1$, the algorithm recovers EXTRA as highlighted in Remark **??**. In this case, our step-size condition is on the order of $O((1 - \sigma_{\max})/\delta)$. Note that the in the original EXTRA proof in [5, Theorem 3.7], the step-size bound is on the order of $O(\nu_\rho(1 - \sigma_{\max})/\delta^2))$, which scales badly for ill-conditioned problems, i.e., if δ is much larger than ν_ρ.

References

[1] Nicholas Metropolis, Arianna W Rosenbluth, Marshall N Rosenbluth, Augusta H Teller, and Edward Teller. Equation of state calculations by fast computing machines. *The Journal of Chemical Physics*, 21(6):1087–1092, 1953.

[2] S Unnikrishna Pillai, Torsten Suel, and Seunghun Cha. The Perron-Frobenius theorem: Some of its applications. *IEEE Signal Processing Magazine*, 22(2):62–75, 2005.

[3] A. H. Sayed. Adaptation, learning, and optimization over neworks. *Foundations and Trends in Machine Learning*, 7(4-5):311–801, 2014.

[4] W. Shi, Q. Ling, K. Yuan, G. Wu, and W. Yin. On the linear convergence of the ADMM in decentralized consensus optimization. *IEEE Trans. Signal Process.*, 62(7):1750–1761, 2014.

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

[6] Lin Xiao and Stephen Boyd. Fast linear iterations for distributed averaging. *Systems & Control Letters*, 53(1):65–78, 2004.

Acknowledgements and Contact Information

- This work was supported in part by NSF grant CCF-1524250.
- Emails: slaghunaim@ucla.edu, kunyuan@ucla.edu, and ali.sayed@epfl.ch