An Accelerated Method for Decentralized Distributed Optimization on Time-Varying Networks

Alexander Rogozin

Moscow Institute of Physics and Technology

11 October 2018

Machine Learning Motivation

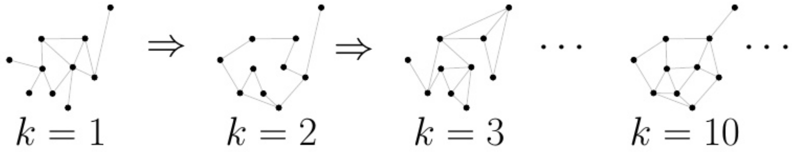
Consider a machine learning problem with a vector of parameters $y \in \mathbb{R}^d$ and a loss function $L(\mathbf{A}, y)$, where $\mathbf{A}$ is a training set of l samples, and each sample is a vector of $\mathbb{R}^m$. The dataset is divided into n parts $\mathbf{A}_i$ and placed on n different machines.

$$L(\mathbf{A}, y) = \sum_{i=1}^{n} L(\mathbf{A_i}, y) \longrightarrow \min_{y \in \mathbb{R}^d} \tag{1}$$

$$\varphi(y) = \sum_{i=1}^{n} \varphi_i(y) \longrightarrow \min_{y \in \mathbb{R}^d} \tag{2}$$

Time-Varying Network

Time-varying network is represented by a sequence of graphs $\{\mathcal{G}_k\}_{k=1}^{\infty}$, where every $\mathcal{G}_k = (V, E_k)$ is a connected undirected graph.



Communication Matrix

Definition

Let $\mathcal{G} = (V, E)$ be a connected undirected graph. Then its Laplacian is defined as

$$[W]_{ij} = \begin{cases} -1, & \text{if } (i, j) \in E, \\ \deg(i), & \text{if } i = j, \\ 0, & \text{otherwise} \end{cases}$$

Basic properties :

- W and $\sqrt{W}$ are symmetric and positive semidefinite
- Vector $\mathbf{1}$ is the unique (up to a scaling factor) eigenvector associated with the eigenvalue $\lambda = 0$

Communication Matrix

Definition

Let $\mathcal{G} = (V, E)$ be a connected undirected graph. Then its Laplacian is defined as

$$[W]_{ij} = \begin{cases} -1, & \text{if } (i,j) \in E, \\ \deg(i), & \text{if } i = j, \\ 0, & \text{otherwise} \end{cases}$$

Basic properties :

- W and $\sqrt{W}$ are symmetric and positive semidefinite
- Vector $\mathbf{1}$ is the unique (up to a scaling factor) eigenvector associated with the eigenvalue $\lambda = 0$

Reformulation via Communication Matrix

Problem

$$\varphi(y) = \sum_{i=1}^{n} \varphi_i(y) \longrightarrow \min_{y \in \mathbb{R}^d} \tag{3}$$

can be equivalently rewritten as

$$\sum_{i=1}^{n} \varphi_i(y_i) \longrightarrow \min_{\substack{y_1 = \ldots = y_n \\ y_i \in \mathbb{R}^d}} \tag{4}$$

or, using Laplacian properties,

$$\Phi(Y) = \sum_{i=1}^{n} \varphi_i(y_i) \longrightarrow \min_{Y\sqrt{W}=0} \tag{5}$$

where we denote $Y = [y_1 \; \ldots \; y_n] \in \mathbb{R}^{d \times n}$. This brings us to the minimization problem

$$f(X) = \max_{Y \in \mathbb{R}^{d \times n}} \left[-\langle X, Y\sqrt{W} \rangle - \Phi(Y) \right] \longrightarrow \min_{X \in \mathbb{R}^{d \times n}} \tag{6}$$

Reformulation via Communication Matrix

Problem

$$\varphi(y) = \sum_{i=1}^{n} \varphi_i(y) \longrightarrow \min_{y \in \mathbb{R}^d} \tag{3}$$

can be equivalently rewritten as

$$\sum_{i=1}^{n} \varphi_i(y_i) \longrightarrow \min_{\substack{y_1 = \ldots = y_n \\ y_i \in \mathbb{R}^d}} \tag{4}$$

or, using Laplacian properties,

$$\Phi(Y) = \sum_{i=1}^{n} \varphi_i(y_i) \longrightarrow \min_{Y\sqrt{W} = 0} \tag{5}$$

where we denote $Y = [y_1 \ \ldots \ y_n] \in \mathbb{R}^{d \times n}$. This brings us to the minimization problem

$$f(X) = \max_{Y \in \mathbb{R}^{d \times n}} \left[-\langle X, Y\sqrt{W} \rangle - \Phi(Y) \right] \longrightarrow \min_{X \in \mathbb{R}^{d \times n}} \tag{6}$$

Reformulation via Communication Matrix

Problem

$$\varphi(y) = \sum_{i=1}^{n} \varphi_i(y) \longrightarrow \min_{y \in \mathbb{R}^d} \tag{3}$$

can be equivalently rewritten as

$$\sum_{i=1}^{n} \varphi_i(y_i) \longrightarrow \min_{\substack{y_1 = \ldots = y_n \\ y_i \in \mathbb{R}^d}} \tag{4}$$

or, using Laplacian properties,

$$\Phi(Y) = \sum_{i=1}^{n} \varphi_i(y_i) \longrightarrow \min_{Y\sqrt{W} = 0} \tag{5}$$

where we denote $Y = [y_1 \ \ldots \ y_n] \in \mathbb{R}^{d \times n}$. This brings us to the minimization problem

$$f(X) = \max_{Y \in \mathbb{R}^{d \times n}} \left[-\langle X, Y\sqrt{W} \rangle - \Phi(Y) \right] \longrightarrow \min_{X \in \mathbb{R}^{d \times n}} \tag{6}$$

Reformulation via Communication Matrix

Problem

$$\varphi(y) = \sum_{i=1}^{n} \varphi_i(y) \longrightarrow \min_{y \in \mathbb{R}^d} \tag{3}$$

can be equivalently rewritten as

$$\sum_{i=1}^{n} \varphi_i(y_i) \longrightarrow \min_{\substack{y_1 = \ldots = y_n \\ y_i \in \mathbb{R}^d}} \tag{4}$$

or, using Laplacian properties,

$$\Phi(Y) = \sum_{i=1}^{n} \varphi_i(y_i) \longrightarrow \min_{Y\sqrt{W} = 0} \tag{5}$$

where we denote $Y = [y_1 \; \ldots \; y_n] \in \mathbb{R}^{d \times n}$. This brings us to the minimization problem

$$f(X) = \max_{Y \in \mathbb{R}^{d \times n}} \left[-\langle X, Y\sqrt{W} \rangle - \Phi(Y) \right] \longrightarrow \min_{X \in \mathbb{R}^{d \times n}} \tag{6}$$

Reformulation via Communication Matrix

Problem

$$\varphi(y) = \sum_{i=1}^{n} \varphi_i(y) \longrightarrow \min_{y \in \mathbb{R}^d} \tag{3}$$

can be equivalently rewritten as

$$\sum_{i=1}^{n} \varphi_i(y_i) \longrightarrow \min_{\substack{y_1 = \ldots = y_n \\ y_i \in \mathbb{R}^d}} \tag{4}$$

or, using Laplacian properties,

$$\Phi(Y) = \sum_{i=1}^{n} \varphi_i(y_i) \longrightarrow \min_{Y\sqrt{W} = 0} \tag{5}$$

where we denote $Y = [y_1 \ \ldots \ y_n] \in \mathbb{R}^{d \times n}$. This brings us to the minimization problem

$$f(X) = \max_{Y \in \mathbb{R}^{d \times n}} \left[-\langle X, Y\sqrt{W} \rangle - \Phi(Y) \right] \longrightarrow \min_{X \in \mathbb{R}^{d \times n}} \tag{6}$$

Reformulation via Communication Matrix

We define

$$\begin{aligned} Y(X) &= \arg\max_{Y\in\mathbb{R}^{d\times n}} \left[-\langle X, Y\sqrt{W}\rangle - \Phi(Y)\right], \\ Z &= -X\sqrt{W}, \\ \tilde{Y}(Z) &= \arg\max_{Y\in\mathbb{R}^{d\times n}} \left[\langle Z, Y\rangle - \Phi(Y)\right] \\ &= \arg\max_{Y\in\mathbb{R}^{d\times n}} \left[\sum_{i=1}^n (\langle z_i, y_i\rangle - \varphi_i(y_i))\right], \\ \tilde{Y}(Z) &= \left[\tilde{y}_1(z_1), \dots, \tilde{y}_n(z_n)\right] \end{aligned}$$

and it follows that

$$\tilde{Y}(Z) = \tilde{Y}(-X\sqrt{W}) = Y(X).$$

Moreover, the gradient of this dual function is defined as

$$\nabla f(X) = -Y(X)\sqrt{W} = -\tilde{Y}(-X\sqrt{W})\sqrt{W} = -\tilde{Y}(Z)\sqrt{W}$$

Reformulation via Communication Matrix

We define

$$Y(X) = \arg\max_{Y \in \mathbb{R}^{d\times n}} \left[-\langle X, Y\sqrt{W}\rangle - \Phi(Y) \right],$$

$$Z = -X\sqrt{W},$$

$$\tilde{Y}(Z) = \arg\max_{Y \in \mathbb{R}^{d\times n}} \left[\langle Z, Y\rangle - \Phi(Y) \right]$$

$$= \arg\max_{Y \in \mathbb{R}^{d\times n}} \left[\sum_{i=1}^{n} \left(\langle z_i, y_i\rangle - \varphi_i(y_i) \right) \right],$$

$$\tilde{Y}(Z) = \left[\tilde{y}_1(z_1), \dots, \tilde{y}_n(z_n) \right]$$

and it follows that

$$\tilde{Y}(Z) = \tilde{Y}(-X\sqrt{W}) = Y(X).$$

Moreover, the gradient of this dual function is defined as

$$\nabla f(X) = -Y(X)\sqrt{W} = -\tilde{Y}(-X\sqrt{W})\sqrt{W} = -\tilde{Y}(Z)\sqrt{W}$$

Reformulation via Communication Matrix

We define

$$
\begin{aligned}
Y(X) &= \operatorname*{arg\,max}_{Y \in \mathbb{R}^{d\times n}} \left[-\langle X, Y\sqrt{W}\rangle - \Phi(Y) \right], \\
Z &= -X\sqrt{W}, \\
\tilde{Y}(Z) &= \operatorname*{arg\,max}_{Y \in \mathbb{R}^{d\times n}} \left[\langle Z, Y\rangle - \Phi(Y) \right] \\
&= \operatorname*{arg\,max}_{Y \in \mathbb{R}^{d\times n}} \left[\sum_{i=1}^{n} \left(\langle z_i, y_i\rangle - \varphi_i(y_i) \right) \right], \\
\tilde{Y}(Z) &= \left[\tilde{y}_1(z_1), \dots, \tilde{y}_n(z_n) \right]
\end{aligned}
$$

and it follows that

$$
\tilde{Y}(Z) = \tilde{Y}(-X\sqrt{W}) = Y(X).
$$

Moreover, the gradient of this dual function is defined as

$$
\nabla f(X) = -Y(X)\sqrt{W} = -\tilde{Y}(-X\sqrt{W})\sqrt{W} = -\tilde{Y}(Z)\sqrt{W}
$$

Reformulation via Communication Matrix

We define

$$\begin{aligned} Y(X) &= \operatorname*{arg\,max}_{Y \in \mathbb{R}^{d \times n}} \left[-\langle X, Y\sqrt{W} \rangle - \Phi(Y) \right], \\ Z &= -X\sqrt{W}, \\ \tilde{Y}(Z) &= \operatorname*{arg\,max}_{Y \in \mathbb{R}^{d \times n}} \left[\langle Z, Y \rangle - \Phi(Y) \right] \\ &= \operatorname*{arg\,max}_{Y \in \mathbb{R}^{d \times n}} \left[\sum_{i=1}^{n} \left(\langle z_i, y_i \rangle - \varphi_i(y_i) \right) \right], \\ \tilde{Y}(Z) &= \left[\tilde{y}_1(z_1), \ldots, \tilde{y}_n(z_n) \right] \end{aligned}$$

and it follows that

$$\tilde{Y}(Z) = \tilde{Y}(-X\sqrt{W}) = Y(X).$$

Moreover, the gradient of this dual function is defined as

$$\nabla f(X) = -Y(X)\sqrt{W} = -\tilde{Y}(-X\sqrt{W})\sqrt{W} = -\tilde{Y}(Z)\sqrt{W}$$

Distributed Gradient Descent

Specifically, a gradient descent algorithm on this dual function, would be

$$X^{k+1} = X^k + \alpha Y(X^k)\sqrt{W}$$

or equivalently

$$Z^{k+1} = Z^k - \alpha \tilde{Y}(Z^k)W,$$

Note that each of the agents' subproblems

$$\tilde{y}_i(z_i) = \arg\max_{y \in \mathbb{R}^d} \left[\langle y_i, z_i\rangle - \varphi_i(y_i)\right] \tag{7}$$

can be computed locally.

Require: Each agent $i \in V$ locally holds $\varphi_\mathbf{i}$, z_i and some iteration number K.

for $k = 0, 1, 2, \cdots, K-1$ **do**

1. Solve subproblem in Eq. (7) and obtain $\tilde{y}_i(z_i^k)$.
2. Send $\tilde{y}_i(z_i^k)$ to every neighbor and receive $\tilde{y}_i(z_i^k)$ from every neighbor.
3. Take a gradient step.

end for

Distributed Gradient Descent

Specifically, a gradient descent algorithm on this dual function, would be

$$X^{k+1} = X^k + \alpha Y(X^k)\sqrt{W}$$

or equivalently

$$Z^{k+1} = Z^k - \alpha \tilde{Y}(Z^k)W,$$

Note that each of the agents' subproblems

$$\tilde{y}_i(z_i) = \arg\max_{y \in \mathbb{R}^d} \left[\langle y_i, z_i \rangle - \varphi_i(y_i)\right] \tag{7}$$

can be computed locally.

Require: Each agent $i \in V$ locally holds $\varphi_{\mathbf{i}}$, z_i and some iteration number K.

for $k = 0, 1, 2, \cdots, K-1$ **do**

1. Solve subproblem in Eq. (7) and obtain $\tilde{y}_i(z_i^k)$.
2. Send $\tilde{y}_i(z_i^k)$ to every neighbor and receive $\tilde{y}_i(z_i^k)$ from every neighbor.
3. Take a gradient step.

end for

Connection of graph and dual function properties

Theorem

Let $\sigma_{\max}(W)$ be the largest eigenvalue and $\tilde{\sigma}_{\min}(W)$ be the least nonzero eigenvalue of $W^T W = W^2$, where W is the Laplacian of the communication graph $\mathcal{G} = (V, E)$. Let $\Phi(Y)$ be L_Φ-smooth and μ_Φ-strongly convex w.r.t. $\|\cdot\|_F$. Then $f(X) = \max\limits_{Y \in \mathbb{R}^{d\times n}} \Big(-\langle X\sqrt{W}, Y\rangle - \Phi(Y)\Big)$ is strongly convex with constant $\mu_f = \dfrac{\sqrt{\tilde{\sigma}_{\min}(W)}}{L_\Phi}$ on the subspace $(\mathrm{Ker}\, W)^\perp$ and smooth with constant $L_f = \dfrac{\sqrt{\sigma_{\max}(W)}}{\mu_\Phi}$ on $\mathbb{R}^{d\times n}$.

Time-Varying Setting

When the network topology changes, the Laplacian matrix of the graph changes as well, which defines a sequence of graph Laplacians $\{W_k\}_{k=1}^{\infty}$. As a result, contrary to the fixed network setup, we work with a sequence of dual functions $f_k(x)$, such that

$$f_k(X) = \Phi^*(-X\sqrt{W_k}) = \max_{Y \in \mathbb{R}^{d \times n}} \left(-\langle X, Y\sqrt{W_k} \rangle - \Phi(Y) \right). \qquad (8)$$

Assuming that, even though the network changes with time, the network remains connected. Then, all W_k have the same nullspace :

$$\mathrm{Ker}(W_k) = \{y_1 = ... = y_n\} = \mathrm{Ker}(\sqrt{W_k})$$

Since $\Phi(Y)$ does not change, all $f_k(X)$ have a common point of minimum and the same value of minimum due to the strong duality.

Time-Varying Setting

When the network topology changes, the Laplacian matrix of the graph changes as well, which defines a sequence of graph Laplacians $\{W_k\}_{k=1}^{\infty}$. As a result, contrary to the fixed network setup, we work with a sequence of dual functions $f_k(x)$, such that

$$f_k(X) = \Phi^*(-X\sqrt{W_k}) = \max_{Y\in\mathbb{R}^{d\times n}} \left(-\left\langle X, Y\sqrt{W_k}\right\rangle - \Phi(Y)\right). \qquad (8)$$

Assuming that, even though the network changes with time, the network remains connected. Then, all W_k have the same nullspace :

$$\mathrm{Ker}(W_k) = \{y_1 = \ldots = y_n\} = \mathrm{Ker}(\sqrt{W_k})$$

Since $\Phi(Y)$ does not change, all $f_k(X)$ have a common point of minimum and the same value of minimum due to the strong duality.

Time-Varying Setting

When the network topology changes, the Laplacian matrix of the graph changes as well, which defines a sequence of graph Laplacians $\{W_k\}_{k=1}^{\infty}$. As a result, contrary to the fixed network setup, we work with a sequence of dual functions $f_k(x)$, such that

$$f_k(X) = \Phi^*(-X\sqrt{W_k}) = \max_{Y \in \mathbb{R}^{d\times n}} \left(-\left\langle X, Y\sqrt{W_k}\right\rangle - \Phi(Y)\right). \tag{8}$$

Assuming that, even though the network changes with time, the network remains connected. Then, all W_k have the same nullspace :

$$\mathrm{Ker}(W_k) = \{y_1 = \ldots = y_n\} = \mathrm{Ker}(\sqrt{W_k})$$

Since $\Phi(Y)$ does not change, all $f_k(X)$ have a common point of minimum and the same value of minimum due to the strong duality.

Distributed Nesterov Method

Consider fast gradient method

$$y_{k+1} = x_k - \frac{1}{L}\nabla f_k(x_k), \tag{9a}$$

$$x_{k+1} = \left(1 + \frac{\sqrt{\kappa}-1}{\sqrt{\kappa}+1}\right) y_{k+1} - \frac{\sqrt{\kappa}-1}{\sqrt{\kappa}+1} y_k, \tag{9b}$$

with initial points $y_0 = x_0$ and $\kappa = \frac{L}{\mu}$. Its distributed version is the following :

Require: Each agent $i \in V$ locally holds φ_i and some iteration number N.

Choose $\tilde{z}_0^i = z_0^i$ for all $i \in V$

for $k = 0, 1, 2, \cdots, N-1$ **do**

$\quad \tilde{y}_i(z_i^k) = \arg\max_{y \in \mathbb{R}^d} \left[\langle z_i^k, y\rangle - \varphi_i(y_i)\right]$

$\quad$ Send $\tilde{y}_i(z_i^k)$ to every neighbor and receive $\tilde{y}_j(z_j^k)$ from every neighbor.

$\quad \tilde{z}_i^{k+1} = z_i^k - \frac{1}{L}\sum_{j=1}^{n}[W_k]_{ij}\tilde{y}_j(z_j^k)$

$\quad z_i^{k+1} = \left(1 + \frac{\sqrt{\kappa}-1}{\sqrt{\kappa}+1}\right)\tilde{z}_i^{k+1} - \frac{\sqrt{\kappa}-1}{\sqrt{\kappa}+1}\tilde{z}_i^k$

end for

Distributed Nesterov Method

Consider fast gradient method

$$y_{k+1} = x_k - \frac{1}{L}\nabla f_k(x_k), \tag{9a}$$

$$x_{k+1} = \left(1 + \frac{\sqrt{\kappa}-1}{\sqrt{\kappa}+1}\right) y_{k+1} - \frac{\sqrt{\kappa}-1}{\sqrt{\kappa}+1} y_k, \tag{9b}$$

with initial points $y_0 = x_0$ and $\kappa = \frac{L}{\mu}$. Its distributed version is the following :

Require: Each agent $i \in V$ locally holds φ_i and some iteration number N.

1: Choose $\tilde{z}_0^i = z_0^i$ for all $i \in V$

2: **for** $k = 0, 1, 2, \cdots, N-1$ **do**

3: $\quad \tilde{y}_i(z_i^k) = \arg\max_{y \in \mathbb{R}^d} \left[\langle z_i^k, y\rangle - \varphi_i(y_i)\right]$

4: $\quad$ Send $\tilde{y}_i(z_i^k)$ to every neighbor and receive $\tilde{y}_j(z_j^k)$ from every neighbor.

5: $\quad \tilde{z}_i^{k+1} = z_i^k - \frac{1}{L}\sum_{j=1}^{n}[W_k]_{ij}\tilde{y}_j(z_j^k)$

6: $\quad z_i^{k+1} = \left(1 + \frac{\sqrt{\kappa}-1}{\sqrt{\kappa}+1}\right)\tilde{z}_i^{k+1} - \frac{\sqrt{\kappa}-1}{\sqrt{\kappa}+1}\tilde{z}_i^k$

7: **end for**

Distributed Nesterov Method

Definition

Introduce

$$\theta_{\max} = \sup_{k\geq 0}\left\{\sigma_{\max}(W_k)\right\} < \infty, \tag{10a}$$

$$\theta_{\min} = \inf_{k\geq 0}\left\{\tilde{\sigma}_{\min}(W_k)\right\} > 0. \tag{10b}$$

Then every $f_k(X)$ is μ-strongly convex on $\left(\mathrm{Ker}\ W\right)^{\perp}$ and L-smooth on $\mathbb{R}^n$, where $\mu = \frac{\sqrt{\theta_{\min}}}{L_\Phi}$, $L = \frac{\sqrt{\theta_{\max}}}{\mu_\Phi}$ by Theorem 2.

Distributed Nesterov Method

Definition

Introduce

$$\theta_{\max} = \sup_{k \geq 0} \{\sigma_{\max}(W_k)\} < \infty, \tag{10a}$$

$$\theta_{\min} = \inf_{k \geq 0} \{\tilde{\sigma}_{\min}(W_k)\} > 0. \tag{10b}$$

Then every $f_k(X)$ is μ-strongly convex on $\left(\text{Ker } \; W\right)^{\perp}$ and L-smooth on $\mathbb{R}^n$, where $\mu = \frac{\sqrt{\theta_{\min}}}{L_\Phi}$, $L = \frac{\sqrt{\theta_{\max}}}{\mu_\Phi}$ by Theorem 2.

Main Theorem

Theorem

Let Φ be a μ_Φ-strongly convex L_Φ-smooth function and assume that there is a sequence of undirected connected graphs $\{\mathcal{G}_k\}$ with no more than m changes at the moments $n_1, ..., n_m$. Then, the sequence $\{z_i^k\}$ generated by the distributed Nesterov method has the following property : for any $N > n_m$ it holds that

$$f_N(Z_N) - f^* \leqslant \kappa^m \cdot \frac{L+\mu}{2} \cdot \frac{R^2}{(1+\gamma)^N},$$

where $\theta_{\max}$ and θ_{min} are defined in (10), $L = \frac{\sqrt{\theta_{\max}}}{\mu_\Phi}, \mu = \frac{\sqrt{\theta_{\min}}}{L_\Phi}$, $Z_N = (z_1^N, \cdots, z_n^N)$, $R = \|X_0 - X^*\|_2, \kappa = \frac{L}{\mu}$ and $\gamma = \frac{1}{\sqrt{\kappa}-1}$.

Results

Corollary

Let Φ be a μ_Φ-strongly convex L_Φ-smooth function. Denote $L = \frac{\sqrt{\theta_{\max}}}{\mu_\Phi}, \mu = \frac{\sqrt{\theta_{\min}}}{L_\Phi}$, where $\theta_{\max}, \theta_{min}$ are defined in (10). Assume that there is a sequence of graphs $\{\mathcal{G}_k\}$ with no more than m changes. Then, for any $\varepsilon > 0$, the sequence $\{z_i^k\}$ generated by the distributed Nesterov method has the following property : for any $k \geqslant N+1$, it holds that

$$f_N(Z_k) - f^* \leqslant \varepsilon,$$

where

$$N \geq \sqrt{\kappa} \cdot \log\left(\kappa^m \frac{L+\mu}{2} \frac{R^2}{\varepsilon}\right) = \sqrt{\kappa_\Phi \cdot \chi(W)} \cdot \left(m \log \kappa + \log\left(\frac{L+\mu}{2} \frac{R^2}{\varepsilon}\right)\right),$$

and $\chi(W) = \sqrt{\frac{\theta_{\max}}{\theta_{\min}}}$ is the condition number of the sequence of graphs $\mathcal{G}_k = (V, E_k)$.

Optimality

Nesterov method reaches the optimal iteration complexity of $\Omega(\sqrt{\kappa \cdot \chi(W)} \log \frac{1}{\varepsilon})$ for decentralized algorithms obtained in the paper Bach et al "Optimal Algorithms for smooth and strongly convex distributed optimization in networks", arXiv : 1702.08704, 2017.

Numerical Experiments

The synthetic *rigde regression* problem is defined as

$$\min_{z \in \mathbb{R}^m} \frac{1}{2nl} \|b - Hz\|_2^2 + \frac{1}{2} c \|z\|_2^2. \tag{11}$$

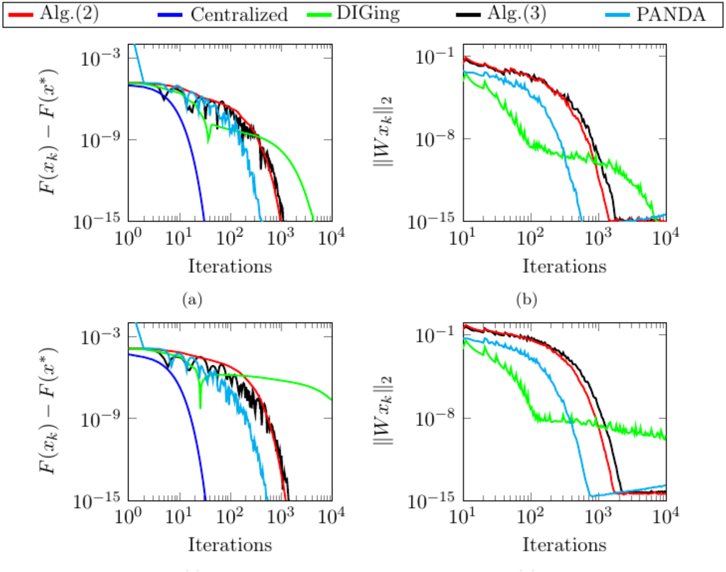
The regularization constant is set to $c = 0.1$. Thus, each agent has access to a subset of points such that

$$b^T = [\ \underbrace{b_1^T}_{\text{Agent } 1}\ |\ \underbrace{b_2^T}_{\text{Agent } 2}\ |\cdots|\ \underbrace{b_n^T}_{\text{Agent } n}\ ] \quad \text{and} \quad H^T = [\ \underbrace{H_1^T}_{\text{Agent } 1}\ |\ \underbrace{H_2^T}_{\text{Agent } 2}\ |\cdots|\ \underbrace{H_n^T}_{\text{Agent } n}\ ],$$

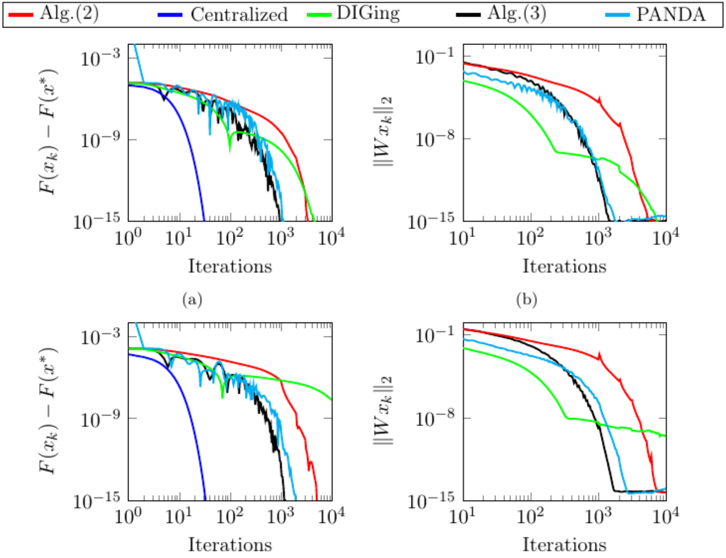
where $b_i \in \mathbb{R}^l$ and $H_i \in \mathbb{R}^{l \times m}$ for each agent $i \in V$. Therefore, in this setup each agent $i \in V$ has a private local function

$$f_i(x_i) \triangleq \frac{1}{2nl} \|b_i - H_i x_i\|_2^2 + \frac{1}{2} \frac{c}{n} \|x_i\|_2^2.$$

Change every 10 iterations



Change every 1000 iterations



References

A. Rogozin, C. A. Uribe, A. Gasnikov, N. Malkovskiy, A. Necich *Optimal Distributed Optimization on Slowly Time-Varying Graphs* arXiv :1805.06045