Decentralized Optimization with Coupled Constraints

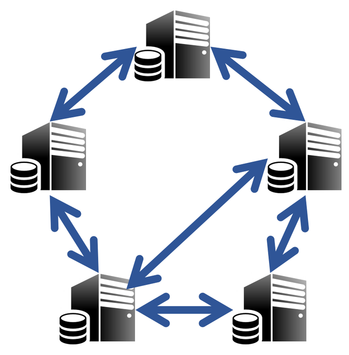
Demyan Yarmoshik [1, 2] Dmitry Kovalev [3] Alexander Rogozin [1, 4] Nikita Kiselev[1] Daniil Dorin[1] Alexander Gasnikov[5,1,2]

[1] Moscow Institute of Physics and Technology [2] Institute for Information Transmission Problems [3] Yandex [4] Skoltech [5] Innopolis University

The problem

We consider the decentralized optimization problem with coupled constraints

$$\min_{x_1\in\mathbb{R}^{d_1},\dots,x_n\in\mathbb{R}^{d_n}} \sum_{i=1}^n f_i(x_i)$$
$$\text{s.t.} \quad \sum_{i=1}^n (\mathbf{A}_i x_i - b_i) = 0$$



Decentralized Topology

Function f_i, matrix $\mathbf{A}_i$ and vector b_i is a private information stored on i-th agent. Agents communicate only with their immediate neighbours in the communication network.

Our goal: obtain a linearly convergent first-order algorithm

Applications

• **Optimal exchange / Resource allocation**

$$\min_{x_1,\dots,x_n\in X} \sum_{i=1}^n f_i(x_i) \quad \text{s.t.} \quad \sum_{i=1}^n x_i = b,$$

where $x_i \in X$ represents the quantities of commodities exchanged among the agents of the system, and $b \in X$ represents the shared budget or demand for each commodity.

• **Problems on graphs.** In electrical microgrids, telecommunication networks, drone swarms, etc, distributed systems are based on physical networks. Electric power network example: let $x_i \in \mathbb{R}^2$ denote the voltage phase angle and the magnitude at i-th electric node, let s be the vector of (active and reactive) power flows for each pair of adjacent electric nodes. Power flows can be derived (with high accuracy) from bus voltages using a linearization of Kirchhoff's law $\sum_{i=1}^n \mathbf{A}_i x_i = s$.

• **Consensus optimization.** Widely used in decentralized machine learning

$$\min_{x_1,\dots,x_n\in X} \sum_{i=1}^n f_i(x_i) \quad \text{s.t.} \quad x_1 = x_2 = \ldots = x_n.$$

The consensus constraint can be reformulated in a decentralized-friendly manner as $\sum_{i=1}^n \mathbf{W}_i x_i = 0$, where $\mathbf{W}_i$ is the i-th vertical block of a gossip matrix (e.g., communication graph's Laplacian).

• **Vertical federated learning (VFL).** Let $\mathbf{F}$ be the matrix of features, split vertically (by features) between agents into submatrices $\mathbf{F}_i$.

$$\min_{\substack{z\in Y\\ x_1\in\mathbb{R}^{d_1},\dots,x_n\in\mathbb{R}^{d_n}}} \ell(z,l) + \sum_{i=1}^n r_i(x_i) \quad \text{s.t.} \sum_{i=1}^n \mathbf{F}_i x_i = z,$$

l is a vector of labels, x_i is a subvector of model parameters owned by the i-th node, ℓ is a loss function, r_i are regularizers.

Assumptions

• All f_i are μ_f-strongly convex and L_f-smooth; $\kappa_f := \frac{L_f}{\mu_f}$.

• The constraints are compatible. There exist constants $L_{\mathbf{A}} \geq \mu_{\mathbf{A}} > 0$, such that the constraint matrices $\mathbf{A}_1,\dots,\mathbf{A}_n$ satisfy $\sigma^2_{\max}(\mathbf{A}) = \max_{i\in 1\dots n}\sigma^2_{\max}(\mathbf{A}_i) \leq L_{\mathbf{A}}$, and $\mu_{\mathbf{A}} \leq \lambda_{\min^+}(\mathbf{S})$, where $\mathbf{S} = \frac{1}{n}\sum_{i=1}^n \mathbf{A}_i\mathbf{A}_i^\top$; $\kappa_{\mathbf{A}} := L_{\mathbf{A}}/\mu_{\mathbf{A}}$.

• We are given a gossip matrix W, such that:

1. $W_{ij} \neq 0$ if and only if agents i and j are neighbours or $i = j$.
2. $Wy = 0$ if and only if $y \in \mathcal{L}_1$, i.e. $y_1 = \ldots = y_n$.
3. There exist constants $L_{\mathbf{W}} \geq \mu_{\mathbf{W}} > 0$ such that $\mu_{\mathbf{W}} \leq \lambda^2_{\min^+}(W)$ and $\lambda^2_{\max}(W) \leq L_{\mathbf{W}}$; $\kappa_{\mathbf{W}} := \frac{\lambda_{\max}(\mathbf{W})}{\lambda_{\min^+}(\mathbf{W})} = \sqrt{\frac{L_{\mathbf{W}}}{\mu_{\mathbf{W}}}}$.

Approach

Decentralized reformulation. Let $\mathbf{A} = \mathrm{diag}\,(A_1,\dots,A_n)$, $\mathbf{b} = (b_1^\top,\dots,b_n^\top)^\top$, $x = (x_1^\top,\dots,x_n^\top)^\top$, $\mathbf{W} = W \otimes I_m$. The original constraint can be equivalently reformulated as $\mathbf{A}x + \gamma\mathbf{W}y = \mathbf{b}$, $\gamma \neq 0$. Matrix multiplications in the reformulation can be performed using single communication with neighbours.

Base algorithm. We use algorithm from [1] (see also [2]), which was proposed for minimization of a smooth strongly convex function $G(u)$ under affine constraint $\mathbf{K}u = \mathbf{b}'$.

Algorithm 1: APAPC

1: $u_g^k := \tau u^k + (1-\tau)u_f^k$
2: $u^{k+\frac{1}{2}} := (1+\eta\alpha)^{-1}(u^k - \eta(\nabla G(u_g^k) - \alpha u_g^k + z^k))$
3: $z^{k+1} := z^k + \theta\mathbf{K}^\top(\mathbf{K}u^{k+\frac{1}{2}} - \mathbf{b}')$
4: $u^{k+1} := (1+\eta\alpha)^{-1}(u^k - \eta(\nabla G(u_g^k) - \alpha u_g^k + z^{k+1}))$
5: $u_f^{k+1} := u_g^k + \frac{2\tau}{2-\tau}(u^{k+1} - u^k)$

This first-order algorithm is based on the Forward-Backward algorithm and Nesterov's acceleration.

Augmentation. In the decentralized reformulation we introduced the variable y, making the objective a *non*-strongly convex function of (x,y). To still obtain linear convergence we add the augmentation term $G(x,y) = \sum_i f_i(x_i) + \frac{r}{2}\|\mathbf{A}x + \gamma\mathbf{W}y - \mathbf{b}\|^2$. With appropriate coefficients, G is smooth and strongly convex enough.

Chebyshev's acceleration. Our constraint matrix $(\mathbf{A}\ \gamma\mathbf{W})$ consists of two matrices, multiplications by which correspond to different oracles. Therefore, we modify application of Chebyshev's acceleration from [1], by replacing $\mathbf{W}$ with $P_W(\mathbf{W})$ first and then applying Chebyshev's acceleration to matrix $(\mathbf{A}\ \gamma P_W(\mathbf{W}))$.

Results

Theorem (Algorithm)

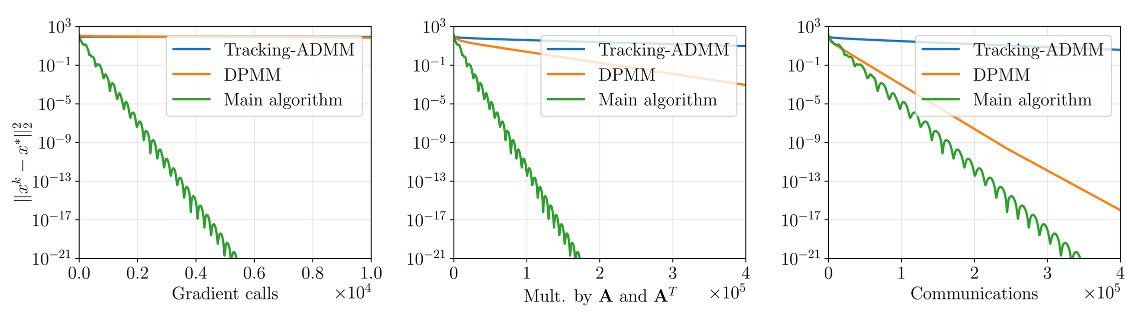
For every $\varepsilon > 0$, the proposed algorithm finds x^k for which $\|x^k - x^*\|^2 \leq \varepsilon$ using $O(\sqrt{\kappa_f}\log(1/\varepsilon))$ objective's gradient computations, $O(\sqrt{\kappa_f}\sqrt{\kappa_{\mathbf{A}}}\log(1/\varepsilon))$ multiplications by $\mathbf{A}$ and $\mathbf{A}^\top$, and $O(\sqrt{\kappa_f}\sqrt{\kappa_{\mathbf{A}}}\sqrt{\kappa_{\mathbf{W}}}\log(1/\varepsilon))$ communication rounds (multiplications by $\mathbf{W}$).

Theorem (Lower bound)

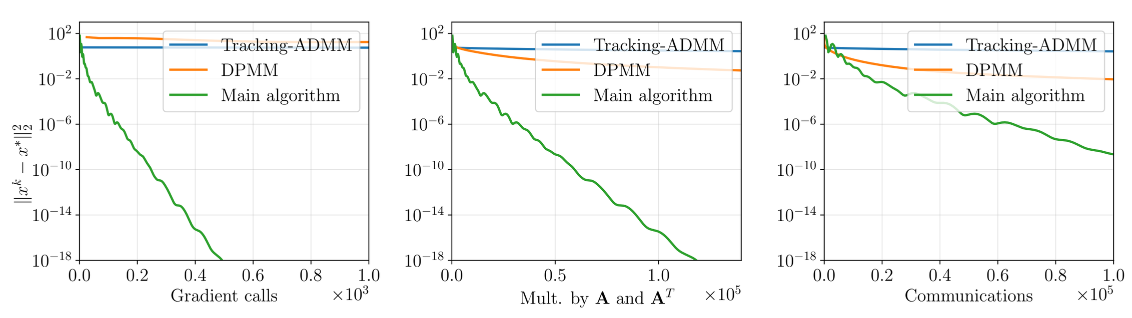
For any $L_f > \mu_f > 0$, $\kappa_{\mathbf{A}}, \kappa_{\mathbf{W}} > 0$ there exist L_f-smooth μ_f-strongly convex functions $\{f_i\}_{i=1}^n$, matrices $\mathbf{A}_i$ such that $\kappa_{\mathbf{A}} = L_{\mathbf{A}}/\mu_{\mathbf{A}}$, and a communication graph $\mathcal{G}$ with a corresponding gossip matrix $\mathbf{W}$ such that $\kappa_{\mathbf{W}} = \lambda_{\max}(\mathbf{W})/\lambda^+_{\min}(\mathbf{W})$, for which any first-order decentralized algorithm to reach accuracy ε requires at least $N_{\mathbf{A}} = \Omega\left(\sqrt{\kappa_f}\sqrt{\kappa_{\mathbf{A}}}\log\left(\frac{1}{\varepsilon}\right)\right)$ multiplications by $\mathbf{A}$ and $\mathbf{A}^\top$ and $N_{\mathbf{W}} = \Omega\left(\sqrt{\kappa_f}\sqrt{\kappa_{\mathbf{A}}}\sqrt{\kappa_{\mathbf{W}}}\log\left(\frac{1}{\varepsilon}\right)\right)$ communication rounds (multiplications by $\mathbf{W}$).

The corresponding lower bound on gradient computations is a classical result by Nesterov.

Experiments



Synthetic VFL, Erdős–Rényi graph, $n = 20$, $d_i = 3$, $m = 10$



LibSVM VFL, Erdős–Rényi graph, $n = 7$, $m = 100$

Summary

The simple augmentation trick and utilization of accelerated Forward-Backward algorithm [2] allowed to overpass the strong convexity issue and obtain an optimal first-order algorithm. Transition to the dual problem was not fruitful in this case.

The analysis is mostly linear algebra to derive spectral properties of block-matrices. All nasty inequalities stuff is hidden in the base algorithm's analysis.

References

[1] Salim et al., An optimal algorithm for strongly convex minimization under affine constraints

[2] Kovalev et al, Optimal and practical algorithms for smooth and strongly convex decentralized optimization