Stochastic mirror descent for fast distributed optimization and federated learning

Anastasia Borovykh, Nikolas Kantas, Panos Parpas, Greg Pavliotis

The setting

In distributed optimization the objective function is given by, $\min_{x\in\mathcal{X}} \sum_{i=1}^{N} f_i(x),$

with $\mathcal{X} \subset \mathbb{R}^d$ a closed convex constraint set and f_i a convex function and N is the total number of nodes in the system. Each node i has access to its local objective function f_i. The communication structure is defined through the underlying communication graph $G:=(V,E)$, where V and E are the vertices and edges, resp. The matrix A represents the communication graph and is assumed to be doubly stochastic.

The goal

Optimizing the objective amounts to finding a solution such that:

1. Consensus holds: $\hat{x}^i = \hat{x}^j$
2. Optimality holds: $\sum_{i=1}^N \nabla f_i(\hat{x}^i) = 0.$

Can we find an algorithm such that both of these objectives are achieved?

The setting

- We work in the **mirror descent** [6] setting with D the Bregman divergence. This setup *can* achieve faster convergence than projected gradient descent due to the ability to adapt to the geometry of the problem.
- Noise is assumed to be additive Brownian and comes from a **noisy gradient estimate or noisy communication**.
- Each node only has access to its *local* objective function f_i and communicates with the other nodes through matrix A.

Algorithm 1.

A standard interacting stochastic mirror descent (ISMD) algorithm for estimating the minimizer is,

$$d\mathbf{z}_t = (-\eta\nabla\mathcal{V}(\mathbf{z}_t) - \epsilon\mathcal{L}\mathbf{z}_t)\,dt + \sigma d\mathbf{B}_t, \quad \mathbf{x}_t = \nabla\Phi^*(\mathbf{z}_t)$$

where

$$\nabla\mathcal{V}_i(z_t^i) := \nabla f_i \circ \nabla\Phi^*(z_t^i) \quad \mathbf{B}_t := ((B_t^1)^T, ..., (B_t^N)^T)^T$$
$$\nabla\mathcal{V}(\mathbf{z}_t) = (\nabla\mathcal{V}_1(z_t^1)^T, ..., \nabla\mathcal{V}_N(z_t^N)^T)^T$$

Will this algorithm converge to consensus and optimality?

The problem

Under the assumptions of smoothness and convexity of f_i it holds,

$$\frac{1}{T}\int_0^T \mathbb{E}\left[(f(x_t^i) - f(x^*))\right] dt \le \frac{C_1}{2T\eta} + \frac{C_2}{\eta}\frac{\sigma^2}{2N} + \frac{C_3\eta}{\lambda\epsilon} + \frac{C_4\sigma}{\sqrt{\lambda\epsilon}},$$

so that:

1. Imposing a small learning rate slows down convergence but allows to converge closer to the optimum if the noise is small or number of particles is big.
2. Imposing a high interaction strength allows to converge closer to the optimum.

Exact convergence is this not achieved due to:

- An additional term arising from the noise,
- An additional term arising from the gradients. This term can only be mitigated by imposing a small learning rate, but this **slows down convergence**!

How can we mitigate this?

Algorithm 2.

We propose an exact algorithm:

$$d\mathbf{v}_t = -\mathcal{L}\mathbf{v}_t dt + \nabla^2 f(\mathbf{x}_t) d\mathbf{x}_t + \sigma d\mathbf{B}_t,$$
$$d\mathbf{z}_t = -\mathcal{L}\mathbf{z}_t dt - \mathbf{v}_t dt,$$

and note that $\nabla^2 f(\mathbf{x}_t) d\mathbf{x}_t = d(\nabla f(\mathbf{x}_t))$, which when discretized yields the update $\nabla f(\mathbf{x}_{t+1}) - \nabla f(\mathbf{x}_t)$.
So what is special here?

- This algorithm incorporates a form of history information.
- Before, the algorithm would be unstable if 1 and 2. was satisfied. Now it is stable.

The

Example 1. A linear system
The exact algorithm converges a lot closer and faster to the optimum. Using a small learning rate or high interaction can help converge closer too.

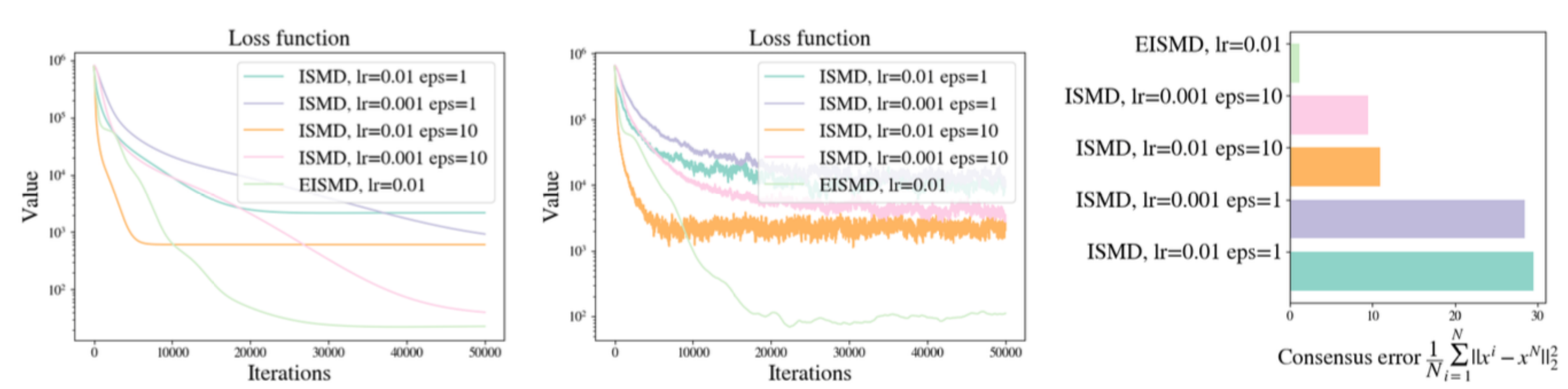

Figure 1: An unconstrained linear system. Comparison of ISMD for different learning rates (lr) and interactions strengths (eps) and EISMD. (L) train loss for $\sigma = 0$, (C) train loss for $\sigma = 0.1$ and (R) the consensus error for $\sigma = 0.1$.

Example 1. A federated learning model.
Theoretically all should work in convex case. But what about the non-convex case where the model is a neural network? We see the exact algorithm performs good too.

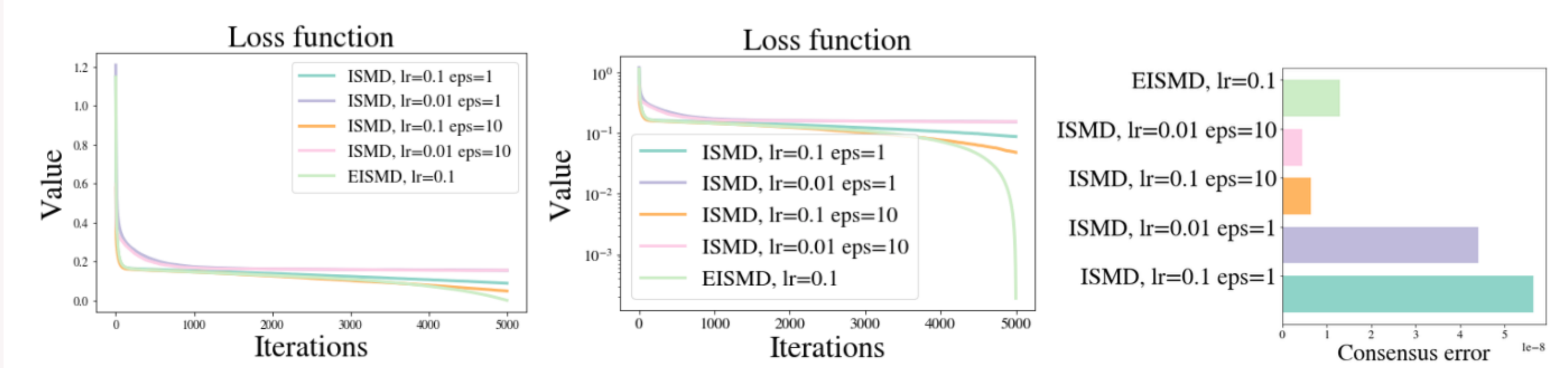

Figure 2: A one-layer neural network with 30 hidden nodes. Comparison of stochastic ISMD for different learning rates (lr) and interactions strengths (eps) and EISMD, both with $\sigma = 0.01$. (L) the average loss on a linear scale, (C) a logarithmic scale and (R) the consensus error.