OPT2020: 12th Annual Workshop on Optimization for Machine Learning

Stochastic mirror descent for fast distributed optimization and federated learning

Anastasia Borovykh A.BOROVYKH@IMPERIAL.AC.UK
Nikolas Kantas N.KANTAS@IMPERIAL.AC.UK
Panos Parpas PANOS.PARPAS@IMPERIAL.AC.UK
Greg Pavliotis G.PAVLIOTIS@IMPERIAL.AC.UK

Abstract

In this work we focus on the convergence properties of interacting stochastic mirror descent (ISMD) in a distributed setting. Our analysis exploits the continuous-time dynamics and explicitly accounts for the noise through additive Brownian motion. We present an overview of necessary conditions for achieving convergence for the ISMD scheme, and show that approximate convergence can be achieved by increasing the interaction strength or decreasing the learning rate. We then present a gradient-tracking ISMD scheme and show that it can converge to the optimal solution without the need of a small learning rate or high interaction strength. We validate our results numerically in a constrained and an unconstrained system and show the potential of the methodology in a federated learning-like scenario using a neural network.

1. Introduction

Using the recently increased computation and storage capabilities of personal devices, machine learning models can be run on-device. This removes the need of transferring data to a central location, reducing infrastructure costs and privacy risks but requires distributed optimization. The goal of distributed optimization is to optimize a sum of different objective functions. Each node the users device will have access to its local objective function and information. A similar setting arises when training a model in a distributed manner and spreading the available data over multiple processors which could be computationally beneficial when having a very large dataset. In such problems the communication or gradient computation may be corrupted, resulting in additional noise being present in the algorithm. These challenges motivate the development of distributed optimization algorithms that are robust to noise and converge quickly.

Distributed optimization is a well-studied problem. Using first-order methods to optimize the global objective by exploiting local communications has been addressed in many works: [16], [5], [3], [10], [17], [15] in discrete time and [4], [6], [19], [8], [20] in continuous time. Typically convergence to a stationary point is shown under the assumption of a vanishing learning rate, which comes at the cost of slowing down convergence. A well-known problem when using a *fixed* learning rate is that of inexact convergence (see e.g. [18]), i.e. the inability to converge exactly to the stationary point. This can be addressed by including 'higher-order' terms in the dynamics [13], [18], [19].

The aim of this work is to study distributed optimization when the objective function is given by

$$\min_{x\in\mathcal{X}} \sum_{i=1}^{N} f_i(x), \tag{1}$$

where $i = 1, ..., N$ are indexing the nodes in the system and $\mathcal{X}$ is a convex constraint set. Let x^* be a minimizer of the objective. Each node has access to its local objective function f_i defined over

$x_i \in \mathcal{X} \subset \mathbb{R}^d$. The communication structure is defined through the underlying communication graph $\mathcal{G} := (V, E)$, where V and E are the vertices and edges, respectively and each node i can communicate only with his direct neighbors $j \in \{j \in V | (i,j) \in E\}$.

In this note we discuss the convergence using the general framework of Mirror Descent [12] in a distributed setting using algorithms in continuous time with additive Brownian noise. Our results for ISMD extend earlier work in [14][2] to the distributed setting. We furthermore sketch extensions for using gradient-tracking ISMD dynamics and show how this can promote consensus in both the loss and the gradients leading to faster convergence to the stationary point. Our analysis allows to obtain quantitative insights into the role of different parameters. We furthermore present numerical experiments that back the analysis as well as experimental evidence that the presented algorithms work well in the non-convex case as encountered in federated learning.

2. The framework

Optimizing the objective in (1) amounts to finding a solution $\hat{x}$ such that the following two conditions are satisfied: 1) consensus: $\hat{x}^i = \hat{x}^j$; 2) optimality: $\sum_{i=1}^N \nabla f_i(\hat{x}^i) = 0$. The matrix $A = \{A_{ij}\}_{i,j=1}^N$ represents the doubly-stochastic communication matrix, with $A_{ij} > 0$ if $(i,j) \in E$. We define the graph Laplacian as $L := \text{Diag}(A\mathbf{1}_N) - A$, and let $\mathcal{L} := L \otimes I_d$, where $\otimes$ is the Kronecker product. We will use a *mirror map* $\Phi : \mathcal{X} \to \mathbb{R}^d$ to convert the constrained optimization problem to an unconstrained one and adopt the following standard assumptions (e.g. [1, Assumption 9.3]) that $\Phi : \mathcal{X} \to \mathbb{R}^d$ is α-strongly convex and continuously differentiable. We furthermore assume that $\nabla\Phi^*(x) = \nabla\Phi^{-1}(x)$.

2.1. Interacting stochastic mirror descent

Define $\nabla\mathcal{V}_i(z_t^i) := \nabla f_i \circ \nabla\Phi^*(z_t^i)$. Consider the following dynamics for interacting stochastic mirror descent (ISMD),

$$d\mathbf{z}_t = (-\eta\nabla\mathcal{V}(\mathbf{z}_t) - \epsilon\mathcal{L}\mathbf{z}_t)\, dt + \sigma d\mathbf{B}_t, \tag{2}$$

where $\mathbf{B}_t := ((B_t^1)^T, ..., (B_t^N)^T)^T$ and $\nabla\mathcal{V}(\mathbf{z}_t) = (\nabla\mathcal{V}_1(z_t^1)^T, ..., \nabla\mathcal{V}_N(z_t^N)^T)^T$. Then $\nabla\Phi^*$ maps $\mathbf{z}$ to the primal space so that $\mathbf{x}_t = \nabla\Phi^*(\mathbf{z}_t)$. Note that $\mathcal{L}$ represents the interaction structure between the nodes. The parameters η, ϵ and σ affect the relative influence of the gradient, interaction and noise, respectively.

2.2. Exact interacting stochastic mirror descent

We define now the exact ISMD (EISMD) algorithm which exploits momentum in order to achieve exact consensus and optimality without requiring a decreasing learning rate. Our algorithm bears similarities to those in [13] and [7]; the novelty here being that we work in continuous time with additional Brownian noise and allow for constraints. The EISMD algorithm is given by,

$$\begin{aligned} d\mathbf{v}_t &= -\mathcal{L}\mathbf{v}_t dt + \nabla^2 f(\mathbf{x}_t) d\mathbf{x}_t + \sigma d\mathbf{B}_t, \\ d\mathbf{z}_t &= -\mathcal{L}\mathbf{z}_t dt - \mathbf{v}_t dt, \\ d\mathbf{x}_t &= \nabla^2\Phi^{-1}(\mathbf{v}_t) d\mathbf{z}_t, \end{aligned} \tag{3}$$

with initial conditions $\mathbf{x}_0 = \mathbf{x}$, $\mathbf{v}_0 = \nabla f(\mathbf{x}_0)$ and $\mathbf{z}_0 = \nabla\Phi(\mathbf{x}_0)$. Note that $\nabla^2 f(\mathbf{x}_t) d\mathbf{x}_t = d\left(\nabla f(\mathbf{x}_t)\right)$, which when discretized yields the update $\nabla f(\mathbf{x}_{t+1}) - \nabla f(\mathbf{x}_t)$. Since (3) involves a Hessian-vector product it can be implemented at the same computational cost as a first-order method.

3. A brief convergence analysis

Let $\bar{z}_t^N := \frac{1}{N}\sum_{i=1}^N z_t^i$, $y_t^N = \nabla\Phi^*(z_t^N)$, $z^* = \nabla\Phi(x^*)$. Let $||\cdot||$ be an arbitrary norm. Suppose, $||\mathcal{L}||_2 \leq \underline{\lambda}$. Assume furthermore that the functions f and $\mathcal{V}$ are β-smooth, L-Lipschitz and strongly convex.

3.1. Conditions for exact convergence of ISMD

Consider the dynamics in (2) with $\sigma = 0$. Assume that $\bigcap_{i=1}^N \{\nabla f_i(x) = 0\} \neq \emptyset$; specifically, let x^* be such that $\nabla f_i(x^*) = 0$ for all $i = 1, ..., N$. Then $\lim_{t\to\infty} x_t^i = x^*$. This result states that the algorithm in (2) can converge *exactly* if $\sigma = 0$ and an x^* exists which simultaneously minimizes all f_i's. If this is not the case then only approximate convergence and consensus can be achieved with the first-order dynamics. The proof will appear elsewhere and relies on the analysis of the candidate Lyapunov function $V_t = \max_{i=1,...,N} V_t^i$ with $V_t^i = D_\Phi(x^*, x_t^i)$ (see eg. Theorem 1 in [18] for the Euclidean case).

3.2. Approximate convergence of ISMD

We now present a result for achieving approximate convergence of (2) by controlling the hyperparameters of the algorithm. Under the assumptions of smoothness and convexity, for $f(x_t) = \sum_{i=1}^N f_i(x_t^i)$ it holds,

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

where C_i can depend on d, the Lipschitz constant of f and the choice of mirror map Φ. The proof of the statement consists of the following steps (see also e.g. Prop. 14 and 17 in [2]): 1) bound $f(x_t^i) - f(x^*)$ by a consensus term $||z_t^i - \bar{z}_t^N||$ and an optimality term $\frac{1}{N}\sum_{i=1}^N (y_t^N - x^*)^T \nabla f_i(x_t^i)$; 2) the consensus term can be bounded by using a Lyapunov function of the form $V_t = (z_t^i - \bar{z}_t^N)^T(z_t^i - \bar{z}_t^N)$ and using the boundedness of the gradients of f; 3) the optimality term can be bounded by using a Lyapunov function of the form $V(\bar{z}_t^N) = D_\Phi(\bar{z}_t^N, z^*)$ in combination with the dynamics of $\bar{z}_t^N$.

From the above result we observe that: 1) imposing a small learning rate slows down convergence significantly, but allows to converge closer to the optimum; 2) imposing a high interaction strength allows to converge closer to the optimum, however in the discretized version of the algorithm a too high interaction strength can result in divergence.

3.3. An analysis of convergence of EISMD

The statement in Section 3.2 used the *boundedness of the gradients*. The motivation for the exact ISMD (3) is to additionally use the smoothness in combination with gradient-tracking to obtain fast convergence [13]. The proof is based on a coupling between consensus and optimality; this differs from the bound in the previous section where consensus and optimality were bounded separately. One possible way to analyze the dynamics in (3) is to rewrite the algorithm as an Augmented Lagrangian method. We note that this connection is possible when the Laplacian matrix is symmetric and static. In the standard gradient descent setting (i.e. when $\Phi(x) = 1/2\|x\|^2$) this connection is well known (see e.g. [11], [4]). Here we extend it to the mirror descent setting; consider the following Dual Augmented Lagrangian, $L(\mathbf{z}, \boldsymbol{\lambda}) = f(\nabla\Phi^*(\mathbf{z})) + (\mathcal{L}\mathbf{z}, \boldsymbol{\lambda})_z^{\mathcal{H}^*} + \|\mathcal{L}^{\frac{1}{2}}\mathbf{z}\|_z^2$, where $(u, v)_{\mathcal{Z}}^{\mathcal{H}^*} = \langle\nabla^2\Phi^*(z)u, v\rangle$ and $\|u\|_z^2 = \langle\nabla^2\Phi^*(z)u, u\rangle$. Augmented Lagrangian methods proceed by a descent step in $\mathbf{z}$ and an ascent step in $\boldsymbol{\lambda}$. It can be shown that this primal-dual algorithm is equivalent to the dynamics in (3). Convergence of the algorithm can then be shown using a Lyapunov function of the form $V_t = ||\mathbf{z}_t - \mathbf{z}^*||_2^2 + ||\boldsymbol{\lambda} - \boldsymbol{\lambda}^*||_2^2$. The full analysis will appear elsewhere.

4. Numerical results

We will highlight the numerical differences between 1) ISMD with increased interaction strength, 2) ISMD with small learning rate, and 3) EISMD. In all settings we consider 10 nodes and the connectivity graph is set to a cyclic graph with each node connected to the previous and next node. The mirror map is set to be the negative entropy function $\Phi(x) = \sum_{j=1}^{d} [x]_j \log([x]_j)$.We use the Euler discretization of the algorithms in (2) and (3).

4.1. Unconstrained linear system

Consider a linear system without constraints similar to [19]. Let $N = 10$ and $d = 100$. We first generate a d-dimensional vector $u \sim \mathcal{N}(10 \times \mathbf{1}_d, I_d)$, and perturb it by $w_i \sim \mathcal{N}(0, I_d)$ so that $u_i = u + w_i$ for $i = 1, ..., N$. $A_i \in \mathbb{R}^{20\times 100}$ is a random matrix with condition number 15, and $b_i \in \mathbb{R}^{20}$ is defined as $b_i = A_i u_i$. Then the local functions are $f_i(x) = \frac{1}{2}||A_i x - b_i||_2^2$ for $i = 1, ..., N$ and $f(x) = \frac{1}{2}||Ax - b||_2^2$. Then $x^* = (A^T A)^{-1} A^T b$. We run the optimization algorithms in (2) and (3) with each x_0^i a feasible randomly generated vector. We set $\Delta t = 0.01$ and let the number of iterations be 50,000. The full code is given in a Google Colab notebook.[1]. We compare the performance of the algorithm for the deterministic setting with $\sigma = 0$ and the stochastic one with $\sigma = 0.1$. In both the deterministic and stochastic case ISMD with high interaction strength converges fast initially but is not able to converge to the optimum, while EISMD is able to converge to optimality. A low learning rate and high interaction strength for ISMD results in similar closeness to optimality as EISMD in the deterministic case, but in the presence of noise EISMD outperforms the rest.

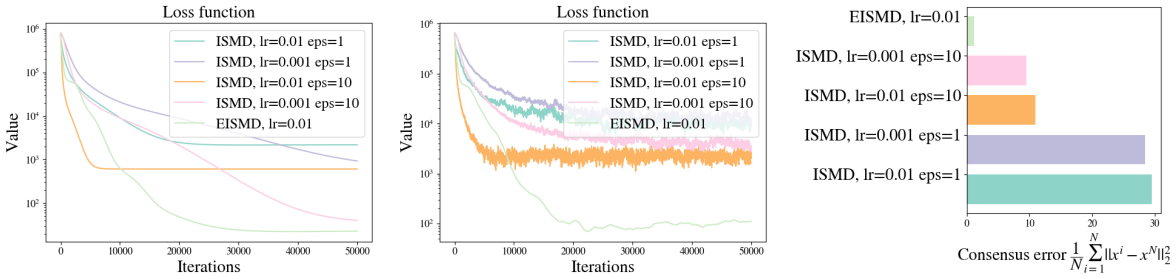

Figure 1: An unconstrained linear system. Comparison of ISMD for different learning rates (lr) and interactions strengths (eps) and EISMD. (L) train loss for $\sigma = 0$, (C) train loss for $\sigma = 0.1$ and (R) the consensus error for $\sigma = 0.1$.

4.2. Simplex constrained linear system

Here we consider a linear setup as in Section 4.1 but set $\mathcal{X} = \Delta_d$, the d-dimensional simplex. The mapping onto the simplex is done using the normalization of the entropy mirror map. We set $\Delta t = 0.01$ and let the number of iterations be 100,000. The left plot in Figure 2 shows the deterministic algorithm with $\sigma = 0$. The EISMD algorithm converges the fastest and achieves the lowest loss value. The center plot in Figure 2 shows the stochastic algorithm with $\sigma = 0.01$. The EISMD algorithm again converges fast to a low loss value; it is noteworthy that the performance of the ISMD algorithm with low learning rate and high interaction strength is similar to that of EISMD. The right plot shows the consensus error which is lowest for a high interaction and low learning rate and EISMD.

1. https://colab.research.google.com/drive/1rvbM7alM5OAKJCgYvmCFaF0yqpGqLeF8?usp=sharing

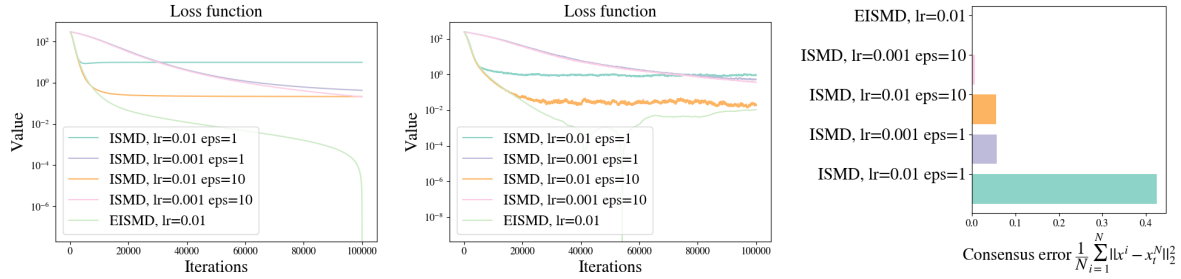

Figure 2: A constrained linear system. Comparison of ISMD for different learning rates (lr) and interactions strengths (eps) and EISMD. (L) train loss for $\sigma = 0$, (C) train loss for $\sigma = 0.1$, (R) the consensus error for $\sigma = 0.1$.

4.3. Federated learning

In standard federated learning [9] applications the underlying model is a deep neural network, which results in a nonconvex objective. In this example we study the performance of ISMD and EISMD in a simple neural network to showcase the potential of the methods in non-trivial settings; this analysis forms the basis for the methodology to be used in FL settings. The neural network architecture consists of one hidden layer with 30 nodes per layer, with a sigmoid activation, a softmax output and the cross-entropy loss. The train dataset is a subset (1000 samples) of the FashionMNIST data. Each node has access to 100 of these samples; the cross-entropy loss over this subset is then the local objective function. The full code is given in a Google Colab notebook.[2] We set $\Delta t = 0.05$ and let the number of iterations be 5,000.

We compare the performance of the algorithms in the stochastic setting with $\sigma = 0.01$ in Figure 3, with the left and center plots showing the average loss over all nodes with each individual loss computed using the parameters x_t^i at node i over the *full* train data, and the right plot showing the consensus error. The EISMD algorithm converges at a similar speed as the algorithms with high learning rate (ISMD with $\eta = 0.1$), however even in the noisy regime EISMD is able to converge to a lower cross-entropy loss value. Similarly the consensus error is, as expected, lowest for high interaction strength or EISMD. These results are especially remarkable since they show the potential of the methodologies – both ISMD with high interaction strength and EISMD – in distributed optimization over non-convex loss surfaces.

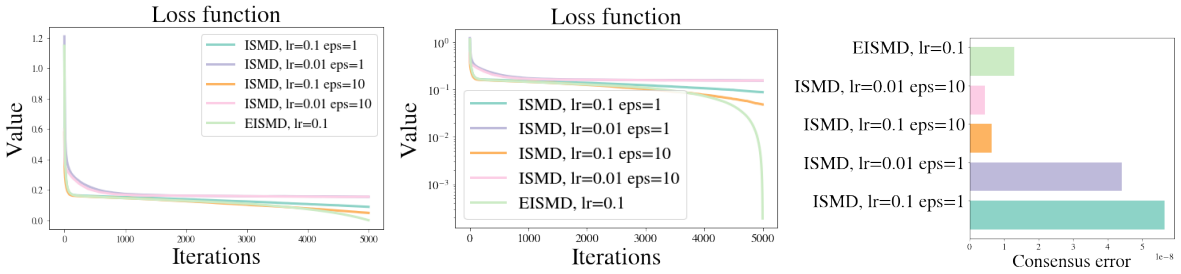

Figure 3: A one-layer neural network with 30 hidden nodes. Comparison of stochastic ISMD for different learning rates (lr) and interactions strengths (eps) and EISMD, both with $\sigma = 0.01$. (L) the average loss on a linear scale, (C) a logarithmic scale and (R) the consensus error.

5. Conclusion

In this work we presented an analysis of distributed stochastic mirror descent using a first-order and second-order optimization scheme and analyzed the convergence. Using numerical results we showed the promising performance of the algorithms in both convex and non-convex objectives.

2. https://colab.research.google.com/drive/1RYLXbGB2zjXAIwy0ax4n17X2MR75Gx-Y?usp=sharing

References

[1] Amir Beck. *First-order methods in optimization*, volume 25. SIAM, 2017.

[2] Anastasia Borovykh, Parpas Parpas, Nikolas Kantas, and Grigorios A. Pavliotis. On stochastic mirror descent with interacting particles: convergence properties and variance reduction. *Submitted to Physica D.*, 2020.

[3] John C Duchi, Alekh Agarwal, and Martin J Wainwright. Dual averaging for distributed optimization: Convergence analysis and network scaling. *IEEE Transactions on Automatic control*, 57(3): 592–606, 2011.

[4] Bahman Gharesifard and Jorge Cortés. Distributed continuous-time convex optimization on weight-balanced digraphs. *IEEE Transactions on Automatic Control*, 59(3):781–786, 2013.

[5] Dušan Jakovetić, Joao Xavier, and José MF Moura. Fast distributed gradient methods. *IEEE Transactions on Automatic Control*, 59(5):1131–1146, 2014.

[6] Peng Lin, Wei Ren, and Jay A Farrell. Distributed continuous-time optimization: nonuniform gradient gains, finite-time convergence, and convex constraint set. *IEEE Transactions on Automatic Control*, 62(5):2239–2253, 2016.

[7] Changxin Liu, Yang Shi, and Huiping Li. Accelerated decentralized dual averaging. *arXiv preprint arXiv:2007.05141*, 2020.

[8] Shuai Liu, Zhirong Qiu, and Lihua Xie. Continuous-time distributed convex optimization with set constraints. *IFAC Proceedings Volumes*, 47(3):9762–9767, 2014.

[9] Brendan McMahan, Eider Moore, Daniel Ramage, Seth Hampson, and Blaise Aguera y Arcas. Communication-efficient learning of deep networks from decentralized data. In *Artificial Intelligence and Statistics*, pages 1273–1282, 2017.

[10] Angelia Nedić, Soomin Lee, and Maxim Raginsky. Decentralized online optimization with global objectives and local communication. In *2015 American Control Conference (ACC)*, pages 4497–4503. IEEE, 2015.

[11] Angelia Nedic, Alex Olshevsky, and Wei Shi. Achieving geometric convergence for distributed optimization over time-varying graphs. *SIAM Journal on Optimization*, 27(4):2597–2633, 2017.

[12] Arkadii Semenovich Nemirovsky and David Borisovich Yudin. Problem complexity and method efficiency in optimization. 1983.

[13] Guannan Qu and Na Li. Harnessing smoothness to accelerate distributed optimization. *IEEE Transactions on Control of Network Systems*, 5(3):1245–1260, 2017.

[14] Maxim Raginsky and Jake Bouvrie. Continuous-time stochastic mirror descent on a network: Variance reduction, consensus, convergence. In *2012 IEEE 51st IEEE Conference on Decision and Control (CDC)*, pages 6793–6800. IEEE, 2012.

[15] Srinivasan Sundhar Ram, Angelia Nedić, and Venugopal V Veeravalli. Distributed stochastic subgradient projection algorithms for convex optimization. *Journal of optimization theory and applications*, 147(3):516–545, 2010.

[16] Kevin Seaman, Francis Bach, Sébastien Bubeck, Yin Tat Lee, and Laurent Massoulié. Optimal algorithms for smooth and strongly convex distributed optimization in networks. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pages 3027–3036. JMLR. org, 2017.

[17] Shahin Shahrampour and Ali Jadbabaie. Distributed online optimization in dynamic environments using mirror descent. *IEEE Transactions on Automatic Control*, 63(3):714–725, 2017.

[18] 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.

[19] Youbang Sun and Shahin Shahrampour. Distributed mirror descent with integral feedback: Asymptotic convergence analysis of continuous-time dynamics. *arXiv preprint arXiv:2009.06747*, 2020.

[20] Xianlin Zeng, Peng Yi, and Yiguang Hong. Distributed continuous-time algorithm for constrained convex optimizations via nonsmooth analysis approach. *IEEE Transactions on Automatic Control*, 62(10):5227–5233, 2016.