Automatica 90 (2018) 196–203

Contents lists available at ScienceDirect

Automatica

journal homepage: www.elsevier.com/locate/automatica

Brief paper

Optimal distributed stochastic mirror descent for strongly convex optimization[☆]

Deming Yuan [a,b,*], Yiguang Hong [c], Daniel W.C. Ho [d], Guoping Jiang [b]

[a] School of Automation, Nanjing University of Science and Technology, Nanjing 210094, Jiangsu, PR China
[b] College of Automation, Nanjing University of Posts and Telecommunications, Nanjing 210023, Jiangsu, PR China
[c] Key Laboratory of Systems and Control, Academy of Mathematics and Systems Science, Chinese Academy of Sciences, Beijing 100190, PR China
[d] Department of Mathematics, City University of Hong Kong, Kowloon, Hong Kong

ARTICLE INFO

Article history:
Received 14 October 2016
Received in revised form 17 June 2017
Accepted 10 November 2017
Available online 15 February 2018

Keywords:
Distributed stochastic optimization
Strong convexity
Non-Euclidean divergence
Mirror descent
Epoch gradient descent
Optimal convergence rate

ABSTRACT

In this paper we consider convergence rate problems for stochastic strongly-convex optimization in the non-Euclidean sense with a constraint set over a time-varying multi-agent network. We propose two efficient non-Euclidean stochastic subgradient descent algorithms based on the Bregman divergence as distance-measuring function rather than the Euclidean distances that were employed by the standard distributed stochastic projected subgradient algorithms. For distributed optimization of non-smooth and strongly convex functions whose only stochastic subgradients are available, the first algorithm recovers the best previous known rate of $O(\ln(T)/T)$ (where T is the total number of iterations). The second algorithm is an epoch variant of the first algorithm that attains the optimal convergence rate of $O(1/T)$, matching that of the best previously known centralized stochastic subgradient algorithm. Finally, we report some simulation results to illustrate the proposed algorithms.

1. Introduction

Recent years have witnessed a growing interest in developing distributed subgradient algorithms for solving convex constrained optimization problem, where the objective function is the sum of the local convex objective functions of nodes in a network (see, e.g. Lin, Ren, & Song, 2016; Nedić, Ozdaglar, & Parrilo, 2010; Zhu & Martínez, 2012), due to their widespread applications including sensor networks (see, e.g. Shi, Ling, Wu, & Yin, 2015), and smart grid (see, e.g. Chang, Nedić, & Scaglione, 2014; Yi, Hong, & Liu, 2016), to name a few.

Strong convexity has been widely studied in convex optimization, because strongly convex cost functions can be easily found in a variety of engineering application domains like sensor networks and smart grids and strongly convex properties are actively used in regularization methods. Take the ridge regression problem as an example, where the objective function consists of the strongly convex Tikhonov regularization term for some performance improvement in optimization computation (see, e.g. Shalev-Shwartz & Ben-David, 2014). In light of the increasing attention to distributed optimization, various distributed designs for optimizing strongly convex functions (in the Euclidean sense) have been proposed in the literature (see Nedić & Olshevsky, 2016 and Tsianos & Rabbat, 2012), due to its wide application in many practical fields and its potential to provide better guarantees of convergence performance.

Many algorithms have been developed over the past years to solve distributed convex optimization problems (see, e.g. Chen & Sayed, 2012; Kia, Cortés, & Martínez, 2015; Liu, Qiu, & Xie, 2014; Lu, Tang, Regier, & Bow, 2011; Ram, Nedić, & Veeravalli, 2010; Yuan, Ho, & Hong, 2016; Yuan, Ho, & Xu, 2016). Such algorithms require only the first-order information of the objective functions and Euclidean projection onto the constraint set. This makes the algorithms attractive for large-scale optimization problems. Specifically, recently an $O(\ln T/\sqrt{T})$ rate of convergence has been established in Nedić and Olshevsky (2015). However, the aforementioned algorithms are inherently Euclidean, in the sense that they rely on measuring distances based on Euclidean norms. This means that it is challenging or infeasible to generate efficient projections for certain objective functions and constraint sets, taking the Euclidean projection onto the unit simplex as an example. In this paper, we shall develop a class of distributed algorithms that are built on mirror descent, which generalizes the projection step using the Bregman divergence. Bregman divergences are a general class of distance-measuring functions, which include the

[☆] The material in this paper was not presented at any conference. This paper was recommended for publication in revised form by Associate Editor Giancarlo Ferrari-Trecate under the direction of Editor Ian R. Petersen.

[*] Corresponding author at: School of Automation, Nanjing University of Science and Technology, Nanjing 210094, Jiangsu, PR China.

E-mail addresses: dmyuan@njupt.edu.cn (D. Yuan), yghong@iss.ac.cn (Y. Hong), madaniel@cityu.edu.hk (D.W.C. Ho), jianggp@njupt.edu.cn (G. Jiang).

https://doi.org/10.1016/j.automatica.2017.12.053
0005-1098/© 2018 Elsevier Ltd. All rights reserved.

Euclidean distance and Kullback–Leibler (KL) divergence as special cases. The work Xi, Wu, and Khan (2014) presents a first study of the distributed optimization algorithm that builds on mirror descent, for solving the non-strongly and deterministic variant of problem (1); however, only convergence results are established for the proposed algorithm.

Convergence rate is an important issue in the distributed design. Although the aforementioned algorithms in the last paragraph can be applied to distributed optimization of strongly convex functions, it is desirable to develop algorithms by further exploiting the strongly convexity of the objective function, in order to provide better performance such as *faster* convergence rates. In Nedić and Olshevsky (2016), the authors proposed a distributed stochastic subgradient-push algorithm for solving problem (1), under the assumption that the stochastic gradients of the objective functions are Lipschitz. In particular, the algorithm converges at an $O(\ln(T)/T)$ rate in the unconstrained case, which is (to the best of our knowledge) the previously best known rate in the literature. The work Rabbat (2015) developed a distributed proximal subgradient algorithm, that uses the Euclidean distance as the distance-measuring function, for solving the unconstrained composite stochastic optimization problems; they prove that the proposed algorithm converges at an $O(1/T)$ rate, under the smoothness assumptions on the objective functions. The authors in Tsianos and Rabbat (2012) proposed a class of distributed algorithms (in both batch and online setting) that converge at an $O(\ln(T)/T)$ rate in the constrained case, without making the smoothness assumptions on the objective functions. Notably, recently the work Lan, Lee, & Zhou (2017) proposed a class of distributed stochastic optimization algorithms that converge at a rate of $O(1/T^2)$, however, note that the algorithms are built on the accelerated subgradient schemes that utilize two previous estimates in the subgradient step.

In this paper we focus on establishing the convergence rate of algorithms for the distributed strongly convex constrained optimization problem in the following form

$$\begin{aligned} &\text{minimize} \quad F(\mathbf{w}) = \sum_{i=1}^{m} F_i(\mathbf{w}) \\ &\text{subject to} \quad \mathbf{w} \in \mathcal{W} \end{aligned} \tag{1}$$

where each F_i is strongly convex in the non-Euclidean sense and maybe non-smooth, and $\mathcal{W} \subseteq \mathbf{R}^d$ is a convex constraint set known to all the nodes in the network. Moreover, the nodes can only compute the noisy subgradients of their respective objective functions. To be specific, we assume that there exists a stochastic subgradient oracle, which, for any point $\mathbf{w} \in \mathcal{W}$, returns a random estimate $\hat{\mathbf{g}}_i(\mathbf{w})$ of a subgradient $\mathbf{g}_i(\mathbf{w}) \in \partial F_i(\mathbf{w})$ so that $\mathbb{E}[\hat{\mathbf{g}}_i(\mathbf{w})] = \mathbf{g}_i(\mathbf{w})$, where $\partial F_i(\mathbf{w})$ denotes the subdifferential set of $F_i(\cdot)$ at $\mathbf{w}$. It is well-known that for (centralized) stochastic optimization of non-smooth and strongly convex functions, the optimal convergence rate is $O(1/T)$ (see, e.g. Hazan & Kale, 2014). This fact, combined with the above observations, motivates us to consider the following questions: (1) Is it possible to develop a distributed stochastic mirror descent algorithm that recovers the best previously known rate $O(\ln(T)/T)$, for distributed optimization of non-smooth and strongly convex functions? and (2) For the same optimization problem, is it possible to devise a variant of the developed algorithm that attains the optimal $O(1/T)$ convergence rate?

In this paper, we give affirmative answers to the above questions. Specifically, the main contributions of this paper are highlighted as follows:

- We consider the construction of non-Euclidean algorithms for distributed stochastic optimization of strongly convex functions whose only stochastic subgradients are available. The algorithms generalize the standard distributed stochastic projected subgradient algorithms to the non-Euclidean setting. Therefore, the proposed algorithms are more flexible, in the sense that they enable us to generate efficient updates to better reflect the geometry of the underlying optimization problem, by carefully choosing the Bregman divergence.
- We propose a distributed stochastic mirror descent (DSMD) algorithm to answer the first question. In particular, we show that for a total number of T iterations, the proposed algorithm achieves an $O(\ln(T)/T)$ rate of convergence, by exploiting the strongly convexity of the objective functions. The DSMD algorithm is a stochastic variant of the algorithm in Xi et al. (2014), where only asymptotic convergence is established. In addition, this rate recovers the best previous known rate in Nedić and Olshevsky (2016) and Tsianos and Rabbat (2012). Moreover, in contrast to the algorithm in Nedić and Olshevsky (2016), our proposed DSMD algorithm is in constrained setting, which naturally arises in a number of applications where each node's estimate has to lie within some decision space (see, e.g. Nedić et al., 2010).
- We propose an epoch variant of the DSMD algorithm, called Epoch-DSMD algorithm, to answer the second question. The Epoch-DSMD algorithm combines the strength of the epoch gradient descent algorithm that is widely used in the machine learning community (see, e.g. Hazan & Kale, 2014) and the DSMD algorithm. In particular, we prove by induction that the resulting point returned by the last epoch attains the optimal $O(1/T)$ rate of convergence, which largely improve the $O(\ln(T)/T)$ rate obtained by Tsianos and Rabbat (2012) with the Euclidean norm.

Notation: Let $\mathbf{R}^d$ be the d-dimensional vector space. Write $\|\mathbf{w}\|_2$ to denote the Euclidean norm of a vector $\mathbf{w} \in \mathbf{R}^d$, and $\langle \mathbf{w}, \mathbf{v} \rangle$ to denote the standard inner product on $\mathbf{R}^d$, for any $\mathbf{w}, \mathbf{v} \in \mathbf{R}^d$. We denote by $[m]$ the set of integers $\{1, \dots, m\}$. For a vector $\mathbf{w}$, we denote its ith component by $[\mathbf{w}]_i$. We denote the (i, j)th element of a matrix $\mathbf{P}$ by $[\mathbf{P}]_{ij}$. For a differentiable function f, Let $\nabla f(\mathbf{w})$ denote the gradient of $f(\cdot)$ at $\mathbf{w}$, and $\mathbb{E}[X]$ denote the expected value of a random variable X.

2. Problem setting and assumptions

In this paper, we are interested in solving convergence rate problems for (1) over a time-varying multi-agent network. Specifically, let $\mathcal{G}(t) = (\mathcal{V}, \mathcal{E}(t), \mathbf{P}(t))$ be a directed graph that represents the nodes' communication pattern at time t, where $\mathcal{V} = \{1, \dots, m\}$ is the node set, $\mathcal{E}(t)$ is the set of activated links at time t, and $\mathbf{P}(t)$ is the communication matrix at time t. We make the following standard assumption on graph $\mathcal{G}(t)$ (see, e.g. Ram et al., 2010; Yuan, Ho, Xu et al., 2016).

Assumption 1. The graph $\mathcal{G}(t) = (\mathcal{V}, \mathcal{E}(t), \mathbf{P}(t))$ satisfies ($t = 1, 2, \dots$):

(a) There exist a scalar $0 < \xi < 1$ such that $[\mathbf{P}(t)]_{ii} \geq \xi$ for all i and t, and $[\mathbf{P}(t)]_{ij} \geq \xi$ whenever $(j, i) \in \mathcal{E}(t)$;
(b) $\mathbf{P}(t)$ is doubly stochastic, i.e., $\sum_{i=1}^{m} [\mathbf{P}(t)]_{ij} = 1$ and $\sum_{j=1}^{m} [\mathbf{P}(t)]_{ij} = 1$ for all i and j;
(c) There exists some positive integer B such that the graph $\left(\mathcal{V}, \bigcup_{t=sB+1}^{(s+1)B} \mathcal{E}(t)\right)$ is strongly connected for every $s \geq 0$.

We now give the definition of the Bregman divergence, which is crucial in developing the algorithms.