Distributed Mirror Descent for Stochastic Learning over Rate-limited Networks

Matthew Nokleby and Waheed U. Bajwa

***Abstract*—We present and analyze two algorithms—termed distributed stochastic approximation mirror descent (D-SAMD) and *accelerated* distributed stochastic approximation mirror descent (AD-SAMD)—for distributed, stochastic optimization from high-rate data streams over rate-limited networks. Devices contend with fast streaming rates by mini-batching samples in the data stream, and they collaborate via distributed consensus to compute variance-reduced averages of distributed subgradients. This induces a trade-off: Mini-batching slows down the effective streaming rate, but may also slow down convergence. We present two theoretical contributions that characterize this trade-off: (i) bounds on the convergence rates of D-SAMD and AD-SAMD, and (ii) sufficient conditions for order-optimum convergence of D-SAMD and AD-SAMD, in terms of the network size/topology and the ratio of the data streaming and communication rates. We find that AD-SAMD achieves order-optimum convergence in a larger regime than D-SAMD. We demonstrate the effectiveness of the proposed algorithms using numerical experiments.**

I. Introduction

Machine learning at its core involves solving stochastic optimization (SO) problems, where the task is to minimize a stochastic loss function having access only to samples from the (unknown) underlying data distribution. The resulting minimizer is then used for tasks such as dimensionality reduction, classification, clustering, etc. The study of SO problems has a long history, with recent results providing optimum convergence rates for convex SO [1], [2]. There has been a recent surge in works on *distributed optimization*, including iterate/dual/subgradient averaging methods [3]–[9], *alternating direction method of multipliers* or *augmented Lagrangian* methods, [10], [11], and approaches to high-performance distributed computing [12], [13].

In this work, we study the problem of *distributed stochastic convex optimization* from *high-rate data streams* over *rate-limited communication links*. This work is motivated by machine learning in networks of sensors and internet-of-things (IoT) devices: as sensors become cheaper, smaller, and more ubiquitous, networks of sensor and IoT devices will generate and collect data at a rate that outstrips the communications throughput of the network. Fast and efficient strategies are needed for machine learning in such networks.

To this end, we present two strategies for distributed SO from fast, streaming data: distributed stochastic approximation mirror descent (D-SAMD) and accelerated distributed stochastic approximation mirror descent (AD-SAMD). These strategies have three main components: (1) devices collect *mini-batches* of samples from their data streams, which reduces the effective streaming rate, (2) devices perform *averaging consensus* on (sub)gradients of the mini-batches to reduce the variance, and (3) devices update search points via stochastic mirror descent on the averaged subgradients [2]. We bound the convergence rates of D-SAMD and AD-SAMD and derive the optimum mini-batch size. We further derive necessary conditions, in terms of network size, topology, and communications rate, for D-SAMD and AD-SAMD to converge as fast as the *centralized* solution. The upshot is that if the communications rate is not too slow, devices can learn via collaboration nearly as quickly as if all the data were co-located. Finally, we demonstrate the effectiveness of D-SAMD and AD-SAMD via numerical experiments.

Matthew Nokleby is with Wayne State University, Detroit, MI (email: matthew.nokleby@wayne.edu), and Waheed U. Bajwa is with Rutgers University, Piscataway, NJ (email: waheed.bajwa@rutgers.edu). The work of WUB was supported in part by the NSF under grant CCF-1453073 and by the ARO under grant W911NF-14-1-0295.

II. Problem Formulation

The objective of this paper is distributed minimization of the stochastic composite function

$$\psi(\mathbf{x}) = E_\xi[\phi(\mathbf{x},\xi)] \triangleq f(\mathbf{x}) + h(\mathbf{x}), \tag{1}$$

where $\mathbf{x} \in X \subset \mathbb{R}^n$, and X is convex and compact. The space $\mathbb{R}^n$ is endowed with an inner product $\langle\cdot,\cdot\rangle$ that need not be the usual one and a norm $\|\cdot\|$ that need not be the one induced by the inner product. The function $f : X \to \mathbb{R}$ is assumed convex with Lipschitz continuous gradients with constant L. The function $h : X \to \mathbb{R}$ is assumed convex and Lipschitz continuous with constant $\mathcal{M}$. The *subdifferentials* of ψ and h are denoted by $\partial\psi(\mathbf{x})$ and $\partial h(\mathbf{x})$, respectively. In the following, the minimizer and the minimum value of ψ are, respectively, denoted by

$$\mathbf{x}^* \triangleq \arg\min_{\mathbf{x}\in X} \psi(\mathbf{x}) \quad \text{and} \quad \psi^* \triangleq \psi(\mathbf{x}^*). \tag{2}$$

A. Distributed Stochastic Composite Optimization

We consider the minimization of $\psi(\mathbf{x})$ over a network of m nodes, represented by the undirected graph $\mathcal{G} = (V, E)$. Nodes minimize ψ collaboratively by exchanging subgradient information with their neighbors at each communications round. Specifically, each node $i \in V$ transmits a message at each communications round to its set of neighbors, defined as

$$\mathcal{N}_i = \{j \in V : (i,j) \in E\}, \tag{3}$$

where $i \in \mathcal{N}_i$. Message passing between nodes takes place without any error or distortion. Further, we constrain the

messages between nodes to be members of the dual space of X and to satisfy causality.

We suppose that each node $i \in V$ queries a first-order stochastic "oracle" at a fixed rate—which may be different from the rate of message exchange—to obtain noisy estimates of the subgradient of ψ. We use 't' to index time according to *data-acquisition* rounds and define $\{\xi_i(t) \in \Upsilon\}_{t\geq 1}$ to be a sequence of independent (with respect to i and t) and identically distributed (i.i.d.) random variables with *unknown* probability distribution $P(\xi)$ that is supported on a subset of the abstract space Υ. At each data-acquisition round t, node i queries the oracle at search point $\mathbf{x}_i(s)$ to obtain the point $G(\mathbf{x}_i(s), \xi_i(t))$ that is a noisy version of the subgradient of ψ at $\mathbf{x}_i(s)$. We use 's' to index time according to *search-point update* rounds, with possibly multiple data-acquisition rounds per search-point update. The reason for allowing the search-point update index s to be different from the data-acquisition index t is to accommodate the setting in which data arrive at faster rate than the rate at which nodes can communicate with each other. Mathematically, the stochastic subgradient $G(\mathbf{x}, \xi)$ is a Borel function that satisfies the following properties:

$$E[G(\mathbf{x}, \xi)] \triangleq \mathbf{g}(\mathbf{x}) \in \partial\psi(\mathbf{x}), \quad \text{and} \tag{4}$$

$$E[\|G(\mathbf{x}, \xi) - \mathbf{g}(\mathbf{x})\|_*^2] \leq \sigma^2, \tag{5}$$

where $\|\cdot\|_*$ denotes the dual norm induced by $\|\cdot\|$ and $\langle\cdot,\cdot\rangle$, and the expectation is with respect to the distribution $P(\xi)$.

B. Mini-batching for Rate-Limited Networks

A common technique to reduce the variance of the (sub)gradient noise and/or reduce the computational burden in centralized SO is to average "batches" of oracle outputs into a single (sub)gradient estimate. In this paper, we employ this technique, called *mini-batching*, in order to reduce the *communications* burden of distributed SO.

We use a simple communications model. Let $\rho > 0$ be the *communications ratio*, i.e., the fixed ratio between the rate of communications and the rate of data acquisition. That is, $\rho \geq 1$ implies nodes engage in ρ rounds of message exchanges for every data-acquisition round. Similarly, $\rho < 1$ means there is one communications round for every $1/\rho$ data-acquisition rounds. We ignore rounding issues for simplicity.

The mini-batching in our distributed problem proceeds as follows. Each mini-batch round spans $b \geq 1$ data-acquisition rounds and coincides with the search-point update round, i.e., each node i updates its search point at the end of a mini-batch round. In each mini-batch round s, each node i uses its current search point $\mathbf{x}_i(s)$ to compute an average of oracle outputs

$$\theta_i(s) = \frac{1}{b} \sum_{t=(s-1)b+1}^{sb} G(\mathbf{x}_i(s), \xi_i(t)). \tag{6}$$

This is followed by each node computing a new search point $\mathbf{x}_i(s+1)$ using $\theta_i(s)$ and messages received from its neighbors.

In the following, we will use the notation $\mathbf{z}_i(s) \triangleq \theta_i(s) - \mathbf{g}(\mathbf{x}_i(s))$. Then, one can show that $E[\|\mathbf{z}_i(s)\|_*^2] \leq C_*\sigma^2/b$, where C_* is a constant that depends on the norm $\|\cdot\|$. We emphasize that the subgradient noise vectors $\mathbf{z}_i(s)$ depend on the search points $\mathbf{x}_i(s)$; we suppress this notation for brevity.

C. Problem Statement

Mini-batching induces a performance trade-off: Averaging reduces subgradient noise and processing time, but it also reduces the rate of search-point updates (and hence, slows down convergence). In order to carry out distributed SA in an order-optimal manner, the nodes collaborate by carrying out $r \geq 1$ rounds of averaging consensus on their mini-batch averages $\theta_i(s)$ in each mini-batch round s. This leads to the constraint $r \leq b\rho$. If communications is faster, or if the mini-batch rounds are longer, nodes can fit in more rounds of information exchange between each mini-batch round. But when ρ is small, the mini-batch size b needed to enable sufficiently many consensus rounds may be so large that the reduction in subgradient noise is outstripped by the reduction in search-point updates and the resulting convergence speed is sub-optimum. Therefore, our main goal is specification of sufficient conditions for ρ such that the resulting convergence speeds of the proposed distributed SA techniques are optimum.

III. Distributed stochastic approximation mirror descent

In this section we present *distributed stochastic approximation mirror descent* (D-SAMD). This algorithm is based upon stochastic approximated mirror descent, which is a generalized version of stochastic subgradient descent.

A. Stochastic Mirror Descent Preliminaries

Stochastic mirror descent, presented in [2], is a generalization of stochastic subgradient descent. This generalization is characterized by a *distance-generating function* $\omega : X \to \mathbb{R}$ that generalizes the Euclidean norm. The distance-generating function must be continuously differentiable and strongly convex with modulus α. We require two measures of the "radius" of X that will arise in the convergence analysis, defined as follows:

$$D_\omega \triangleq \sqrt{\max_{\mathbf{x}\in X} \omega(x) - \min_{\mathbf{x}\in X} \omega(x)}, \quad \Omega_\omega \triangleq \sqrt{\frac{2}{\alpha} D_\omega}.$$

The distance-generating function induces the *prox function*, or the Bregman divergence $V : X \times X \to \mathbb{R}_+$, which generalizes the Euclidean distance:

$$V(\mathbf{x}, \mathbf{z}) = \omega(\mathbf{z}) - (\omega(\mathbf{x}) + \langle \nabla\omega(\mathbf{x}), \mathbf{z} - \mathbf{x}\rangle). \tag{7}$$

The prox function $V(\mathbf{x}, \cdot)$ inherits strong convexity from $\omega(\cdot)$, but it need not be symmetric or satisfy the triangle inequality. We define the *prox mapping* $P_\mathbf{x} : \mathbb{R}^n \to X$ as

$$P_\mathbf{x}(\mathbf{y}) = \arg\min_{\mathbf{z}\in X} \langle \mathbf{y}, \mathbf{z} - \mathbf{x}\rangle + V(\mathbf{x}, \mathbf{z}). \tag{8}$$

The prox mapping generalizes the usual subgradient descent step, in which one minimizes the local linearization of the objective function regularized by the Euclidean distance of the step taken.

To guarantee convergence for our problem, we require that the resulting prox mapping be 1-Lipschitz continuous in $\mathbf{x}, \mathbf{y}$ pairs, i.e.,

$$\|P_\mathbf{x}(\mathbf{y}) - P_{\mathbf{x}'}(\mathbf{y}')\| \leq \|\mathbf{x} - \mathbf{x}'\| + \|\mathbf{y} - \mathbf{y}\prime\|, \forall\, \mathbf{x}, \mathbf{x}', \mathbf{y}, \mathbf{y}' \in \mathbb{R}^n.$$

This condition clearly holds in the Euclidean setting.

B. Description of D-SAMD

Let $\mathbf{W}$ be a symmetric, doubly-stochastic matrix consistent with the network graph $\mathcal{G}$, i.e., $[\mathbf{W}]_{ij} = 0$ if $(i,j) \notin E$. Also set the constant step size $0 < \gamma \leq \alpha/(2L)$. For simplicity, we suppose that there is a predetermined number of data-acquisition rounds T, which leads to $S = T/b$ mini-batch rounds. We detail the steps of D-SAMD in Algorithm 1.

Algorithm 1 Distributed stochastic approximation mirror descent (D-SAMD)

Require: Doubly-stochastic matrix $\mathbf{W}$, step size γ, number of consensus rounds r, batch size b, and stream of mini-batched subgradients $\theta_i(s)$.

for $i = 1 : m$ **do**
  $\mathbf{x}_i(1) \leftarrow \min_{\mathbf{x}\in X} \omega(\mathbf{x})$ ▷ Initialize search points
end for
for $s = 1 : S$ **do**
  $\forall i, \mathbf{h}_i^0(s) \leftarrow \theta_i(s)$, ▷ Get mini-batched subgradients
  for $q = 1 : r$, $i = 1 : m$ **do**
    $\mathbf{h}_i^q(s) \leftarrow \sum_{j\in\mathcal{N}_i} w_{ij}\mathbf{h}_j^{q-1}(s)$ ▷ Consensus rounds
  end for
  for $i = 1 : m$ **do**
    $\mathbf{x}_i(s+1) \leftarrow P_{\mathbf{x}_i(s)}(\gamma \mathbf{h}_i^r(s))$ ▷ Prox mapping
    $\mathbf{x}_i^{\mathrm{av}}(s+1) \leftarrow \frac{1}{s}\sum_{k=1}^{s} \mathbf{x}_i(k)$ ▷ Average iterates
  end for
end for

return $\mathbf{x}_i^{\mathrm{av}}(S+1)$, $i = 1, \ldots, m$

C. Convergence Analysis

The convergence rate of D-SAMD depends on the bias and variance of the approximate subgradient averages $\mathbf{h}_i^r(s)$. In principle, averaging subgradients together reduces the noise variance and speeds up convergence. However, because averaging consensus results in *approximate* averages, each node takes a slightly different mirror prox step and therefore ends up with a different iterate. At each mini-batch round s, nodes then compute subgradients at different search points, leading to bias in the averages $\mathbf{h}_i^r(s)$.

The following result bounds the expected gap to optimality of D-SAMD iterates. The proof involves bounding the bias and variance of subgradient estimates, bounding the distance between iterates at different nodes, and calculating their impact on convergence speed.

Theorem 1: For D-SAMD, the expected gap to optimality at each node i satisfies

$$E[\psi(\mathbf{x}_i^{\mathrm{av}}(S+1))] - \psi^* \leq \frac{2L\Omega_\omega^2}{\alpha S} + \sqrt{\frac{2(4\mathcal{M}^2 + 2\Delta_S^2)}{\alpha S}} + \sqrt{\frac{\alpha}{2}}\frac{\Xi_S D_\omega}{L},$$

where

$$\Xi_s \triangleq (\mathcal{M} + \sigma/\sqrt{b})(1 + m^2\sqrt{C_*}\lambda_2^r)\times ((1+\alpha m^2\sqrt{C_*}\lambda_2^r)^s - 1) + 2\mathcal{M}, \quad \text{and}$$

$$\Delta_s^2 \triangleq 2(\mathcal{M} + \sigma/\sqrt{b})^2((1+\alpha m^2\sqrt{C_*}\lambda_2^r)^s - 1)^2\times (1 + m^4 C_*\lambda_2^{2r}) + 4C_*\sigma^2/(mb) + 4\lambda_2^{2r}C_*\sigma^2 m^2/b + 4\mathcal{M}$$

quantify the moments of the effective subgradient noise. Here, $\lambda_2 \in [0,1)$ denotes the second-largest eigenvalue of $\mathbf{W}$.

The above convergence rate is akin to that provided in [2], with Δ_s^2 taking the role of the subgradient noise variance. The critical question is how fast communications needs to be for order-optimum convergence speed. After S mini-batch rounds, the network has processed mT data samples. By [2], the gap to optimality for centralized SO is no faster than $O((\mathcal{M} + \sigma)/\sqrt{mT})$ if $\sigma^2 > 0$. In the following we state optimality conditions for the convergence rate of D-SAMD.

Corollary 1: The optimality gap for D-SAMD satisfies

$$E[\psi(\mathbf{x}_i^{\mathrm{av}}(t+1))] - \psi^* = O\left(\frac{\mathcal{M}+\sigma}{\sqrt{mT}}\right), \tag{9}$$

provided the mini-batch size b, the communications ratio ρ, the number of users m, and the Lipschitz constant $\mathcal{M}$ satisfy

$$b = \Omega\left(1 + \frac{\log(mT)}{\rho\log(1/\lambda_2)}\right),\ \rho = \Omega\left(\frac{m^{1/2}\log(mT)}{\sigma T^{1/2}\log(1/\lambda_2)}\right),$$
$$m = O(\sigma^2 T),\ \mathcal{M} = O\left(\min\left\{\frac{1}{m}, \frac{1}{\sqrt{m\sigma^2 T}}\right\}\right).$$

IV. Accelerated Distributed Stochastic Approximation Mirror Descent

Here we present *accelerated* distributed stochastic approximation mirror descent (AD-SAMD), which distributes the accelerated stochastic approximation mirror descent proposed in [2]. In centralized settings, accelerated mirror descent achieves a slightly faster convergence rates. In the distributed setting, this allows for more aggressive mini-batching, and order-optimum convergence is possible with smaller rate ρ.

A. Description of AD-SAMD

The setting for AD-SAMD is the same as for D-SAMD, with a distance function $\omega : X \to \mathbb{R}$, its associated prox function $V : X \times X \to \mathbb{R}$, and the resulting (Lipschitz) prox mapping $P_x : \mathbb{R}^n \to X$. We again suppose a mixing matrix $\mathbf{W} \in \mathbb{R}^{m\times m}$ that is symmetric, doubly stochastic, and consistent with $\mathcal{G}$. The main distinction between accelerated and standard mirror descent is that one maintains several distinct sequences of iterate averages. This involves two sequences of step sizes $\beta_s \in [1,\infty)$ and $\gamma_s \in \mathbb{R}$, which are not held constant. We detail the steps of AD-SAMD in Algorithm 2.

B. Convergence Analysis

As with D-SAMD, the convergence analysis relies on bounds on the bias and variance of the averaged subgradients.

Theorem 2: For AD-SAMD, the expected gap to optimality satisfies

$$E[\Psi(\mathbf{x}_i^{\mathrm{ag}}(S+1))] - \Psi^* \leq \frac{8LD_\omega^2}{\alpha S^2} + 4D_\omega\sqrt{\frac{4M + \Delta_S^2}{\alpha S}} + \sqrt{\frac{32}{\alpha}}D_\omega\Xi_S, \tag{10}$$

where

$$\Delta_s^2 = 2(\mathcal{M} + \sigma/\sqrt{b})^2((1 + 2\gamma_s m^2\sqrt{C_*}L\lambda_2^r)^s - 1)^2 + \frac{4C_*\sigma^2}{b}(\lambda_2^{2r}m^2 + 1/m) + 4\mathcal{M},$$

Algorithm 2 Accelerated distributed stochastic approximation mirror descent (AD-SAMD)

Require: Doubly-stochastic matrix $\mathbf{W}$, step size sequences γ_s, β_s, number of consensus rounds r, batch size b, and stream of mini-batched subgradients $\theta_i(s)$.

for $i = 1 : m$ **do**
  $\mathbf{x}_i(1), \mathbf{x}_i^{\mathrm{md}}(1), \mathbf{x}_i^{\mathrm{ag}}(1) \leftarrow \min_{\mathbf{x}\in X} \omega(\mathbf{x})$ ▷ Initialize search points
end for
for $s = 1 : S$ **do**
  for $i = 1 : m$ **do**
    $\mathbf{x}_i^{\mathrm{md}}(s) \leftarrow \beta_s^{-1}\mathbf{x}_i(s) + (1 - \beta_s^{-1})\mathbf{x}_i^{\mathrm{ag}}(s)$
    $\mathbf{h}_i^0(s) \leftarrow \theta_i(s)$ ▷ Get mini-batched subgradients
  end for
  for $q = 1 : r$, $i = 1 : m$ **do**
    $\mathbf{h}_i^q(s) \leftarrow \sum_{j\in\mathcal{N}_i} w_{ij}\mathbf{h}_j^{q-1}(s)$ ▷ Consensus rounds
  end for
  for $i = 1 : m$ **do**
    $\mathbf{x}_i(s+1) \leftarrow P_{\mathbf{x}_i(s)}(\gamma_s \mathbf{h}_i^r(s))$ ▷ Prox mapping
    $\mathbf{x}_i^{\mathrm{ag}}(s+1) \leftarrow \beta_s^{-1}\mathbf{x}_i(s+1) + (1 - \beta_s^{-1})\mathbf{x}_i^{\mathrm{ag}}(s)$
  end for
end for

return $\mathbf{x}_i^{\mathrm{ag}}(S+1)$, $i = 1, \ldots, m$

and

$$\Xi_s = (\mathcal{M} + \sigma/\sqrt{b})(1 + \sqrt{C_*}m^2\lambda_2^r)\times \\ ((1 + 2\gamma_s m^2\sqrt{C_*}L\lambda_2^r)^s - 1) + 2\mathcal{M}.$$

As with D-SAMD, we study the conditions under which AD-SAMD achieves order-optimum convergence speed.

Corollary 2: The optimality gap satisfies

$$E[\psi(\mathbf{x}_i^{\mathrm{ag}}(S+1)] - \psi^* = O\left(\frac{\mathcal{M}+\sigma}{\sqrt{mT}}\right),$$

provided

$$b = \Omega\left(1 + \frac{\log(mT)}{\rho\log(1/\lambda_2)}\right),\ \rho = \Omega\left(\frac{m^{1/4}\log(mT)}{\sigma T^{3/4}\log(1/\lambda_2)}\right) \\ m = O(\sigma^2 T),\ \mathcal{M} = O\left(\min\left\{\frac{1}{m}, \frac{1}{\sqrt{m\sigma^2 T}}\right\}\right).$$

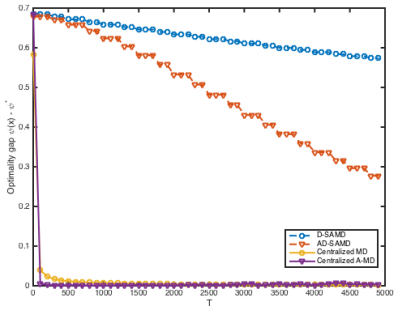
Because AD-SAMD has a convergence rate of $1/S^2$ in the absence of noise and non-smoothness, it tolerates more aggressive mini-batching without impact on the order of the convergence rate. As a result, the condition on ρ is relaxed by $1/4$ in the exponents of m and T.

V. Numerical Example: Logistic Regression

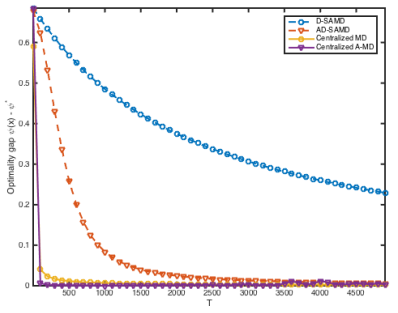
We demonstrate the performance of D-SAMD and AD-SAMD on supervised learning by considering binary logistic regression. A learning machine observes a stream of pairs $\xi(t) = (y(t), l(t))$ of data points $y(t) \in \mathbb{R}^d$ and their labels $l(t) \in \{0, 1\}$, from which it learns a classifier by minimizing the *cross-entropy* loss.

We consider a synthetic example in which the data follow a Gaussian distribution with known, identity covariance but unknown means μ_0 and μ_1. We generate μ_0 and μ_1 randomly from the standard normal distribution. We pick $m = 20$ and draw a graph at random from the Erdős-Rényi distribution with connection probability 0.1. We choose $\mathbf{W}$ to be the Metropolis weights [14] associated with the resulting graph; it turns out here that $\lambda_2 = 0.9436$. For $T = 5000$, we choose step-size $\gamma = 0.005$ and $b = \log(Tm^2)/(\rho\log(1/\lambda_2))$, which guarantees that the equivalent gradient noise variance is $O(\sigma^2/(mT))$.

In Figure 1 we plot the optimality gap for communications ratios $\rho \in \{1, 10\}$. For $\rho = 1$ the mini-batch size b is rather large, so D-SAMD and AD-SAMD only update their search points every few data-acquisition rounds. While one can clearly see the advantage of AD-SAMD over D-SAMD, both centralized algorithms converge much more quickly. By contrast, for $\rho = 10$ the resulting mini-batch size is smaller, and the gap between centralized and decentralized performance is comparable.



(a) $\rho = 1$



(b) $\rho = 10$

Fig. 1: Performance of D-SAMD and AD-SAMD on a synthetic logistic regression problem.

VI. Conclusion

We have presented two distributed schemes, D-SAMD and AD-SAMD, for convex stochastic optimization over networks of nodes that collaborate via rate-limited links. Further, we have derived sufficient conditions for the order-optimum convergence of D-SAMD and AD-SAMD, showing that accelerated mirror descent provides a foundation for distributed SO that better tolerates slow communications links. These results characterize relationships between network communications speed and the convergence speed of stochastic optimization.

REFERENCES

[1] A. Juditsky, A. Nemirovski, and C. Tauvel, "Solving variational inequalities with stochastic mirror-prox algorithm," *Stochastic Systems*, vol. 1, no. 1, pp. 17–58, 2011.

[2] G. Lan, "An optimal method for stochastic composite optimization," *Mathematical Programming*, vol. 133, no. 1, pp. 365–397, 2012.

[3] A. Nedic and A. Ozdaglar, "Distributed subgradient methods for multi-agent optimization," *IEEE Trans. Automat. Control*, vol. 54, no. 1, pp. 48–61, Jan. 2009.

[4] S. Sundhar Ram, A. Nedic, and V. Veeravalli, "Distributed stochastic subgradient projection algorithms for convex optimization," *J. Optim. Theory App.*, vol. 147, no. 3, pp. 516–545, Jul. 2010.

[5] K. Srivastava and A. Nedic, "Distributed asynchronous constrained stochastic optimization," *IEEE J. Select. Topics Signal Processing*, vol. 5, no. 4, pp. 772–790, Aug. 2011.

[6] K. I. Tsianos, S. Lawlor, and M. G. Rabbat, "Push-sum distributed dual averaging for convex optimization," in *Proc. 51st IEEE Conf. Decision and Control (CDC'12)*, Maui, HI, Dec. 2012, pp. 5453–5458.

[7] J. Duchi, A. Agarwal, and M. Wainwright, "Dual averaging for distributed optimization: Convergence analysis and network scaling," *IEEE Trans. Automat. Control*, vol. 57, no. 3, pp. 592–606, Mar. 2012.

[8] A. Mokhtari and A. Ribeiro, "DSA: Decentralized double stochastic averaging gradient algorithm," *J. Mach. Learn. Res.*, vol. 17, no. 61, pp. 1–35, 2016.

[9] A. S. Bijral, A. D. Sarwate, and N. Srebro, "On data dependence in distributed stochastic optimization," *arXiv preprint arXiv:1603.04379*, 2016.

[10] W. Shi, Q. Ling, K. Yuan, G. Wu, and W. Yin, "On the linear convergence of the ADMM in decentralized consensus optimization," *IEEE Trans. Signal Processing*, vol. 62, no. 7, pp. 1750–1761, Apr. 2014.

[11] D. Jakovetić, J. M. F. Moura, and J. Xavier, "Linear convergence rate of a class of distributed augmented lagrangian algorithms," *IEEE Trans. Automat. Control*, vol. 60, no. 4, pp. 922–936, Apr. 2015.

[12] B. Recht, C. Re, S. Wright, and F. Niu, "Hogwild: A lock-free approach to parallelizing stochastic gradient descent," in *Advances in Neural Information Processing Systems*, 2011, pp. 693–701.

[13] H. Mania, X. Pan, D. Papailiopoulos, B. Recht, K. Ramchandran, and M. I. Jordan, "Perturbed iterate analysis for asynchronous stochastic optimization," *arXiv preprint arXiv:1507.06970*, 2015.

[14] L. Xiao, S. Boyd, and S. Lall, "A scheme for robust distributed sensor fusion based on average consensus," in *Proceedings of the 4th International Symposium on Information Processing in Sensor Networks*, 2005, p. 9.