Stochastic Gradient-Push for Strongly Convex Functions on Time-Varying Directed Graphs

Angelia Nedić and Alex Olshevsky

Abstract—We investigate the convergence rate of the recently proposed subgradient-push method for distributed optimization over time-varying directed graphs. The subgradient-push method can be implemented in a distributed way without requiring knowledge of either the number of agents or the graph sequence; each node is only required to know its out-degree at each time. Our main result is a convergence rate of $O\left(\frac{(\ln t)}{t}\right)$ for strongly convex functions with Lipschitz gradients even if only stochastic gradient samples are available; this is asymptotically faster than the $O\left(\frac{(\ln t)}{\sqrt{t}}\right)$ rate previously known for (general) convex functions.

I. INTRODUCTION

We consider the problem of cooperatively minimizing a separable convex function by a network of nodes. Our motivation stems from much recent interest in distributed optimization problems which arise when large clusters of nodes (which can be sensors, processors, autonomous vehicles or UAVs) wish to collectively optimize a global objective by means of actions taken by each node and local coordination between neighboring nodes.

Specifically, we will study the problem of optimizing a sum of $n$ convex functions by a network of $n$ nodes when the $i$th function is known only to node $i$. The functions will be assumed to be from $\mathbb{R}^d$ to $\mathbb{R}$. This problem often arises when control and signal processing algorithms are implemented in sensor networks and global agreement is needed on a parameter which minimizes a sum of local costs. Some specific scenarios in which this problem has been considered in the literature include statistical inference [26], formation control [25], non-autonomous power control [27], distributed “epidemic” message routing in networks [24], and spectrum access coordination [14].

Our focus here is on the case when the communication topology connecting the nodes is time-varying and directed. In the context of wireless networks, time-varying communication topologies arise if the nodes are mobile or if the communication between them is subject to unpredictable bouts of interference. Directed communication links are also a natural assumption as in many cases there is no reason to expect different nodes to transmit wirelessly at the same power level. Transmissions at different power levels will result to unidirectional communication between nodes (usually, however, after an initial exchange of “hello” messages).

In our previous work [22] we proposed an algorithm which is guaranteed to drive all nodes to an optimal solution in this setting. Our algorithm, which we called the subgradient-push, can be implemented in a fully distributed way: no knowledge of the (time-varying) communication topology or even of the total number of nodes is required, although every node is required to know its out-degree at each time. The subgradient-push is a generalization of the so-called push-sum protocol for computing averages on directed graphs proposed over a decade ago [13] (see also the more recent development in [1].

Our main result in [22] was that the subgradient-push protocol drives all the nodes to an optimal solution at a rate $O\left(\frac{(\ln t)}{\sqrt{t}}\right)$. Here, we consider the effect of stronger assumptions on the individual functions. Our main result is that if the functions at each node are strongly convex, and have Lipschitz gradients, then even if each node only has access to noisy gradients of its own function, an improvement to an $O\left(\frac{(\ln t)}{t}\right)$ rate can be achieved.

Our work here contributes to the growing literature on distributed methods for optimization over networks [23], [12], [26], [11], [17], [37], [15], [16], [28], [19], [3], [7], [18], [10], [34]. It is a part of a recent strand of the distributed optimization literature which studies effective protocols when interactions between nodes are unidirectional [4], [55], [4], [9], [8], [80]. Our work is most closely related to recent developments in [30], [31], [32], [9], [29], [56]. We specifically mention [30], [31], which were the first papers to suggest the use of push-sum-like updates for optimization over directed graphs as well as [33], [56] which derived $O(1/t)$ convergence rates in the less stringent setting when every graph is fixed and undirected.

Our paper is organized as follows. In Section II we describe the problem formally and present the algorithm.
along with the main results. The results are then proved in Sections III and IV. We conclude with some simulations in Section V and some concluding remarks in Section VI.

Notation: We use boldface to distinguish between the vectors in $\mathbb{R}^d$ and scalars associated with different nodes. For example, the vector $x_i(t)$ is in boldface to identify a vector for node $i$, while a scalar $y_i(t) \in \mathbb{R}$ is not in boldface. The vectors such as $y(t) \in \mathbb{R}^n$ obtained by stacking scalar values $y_i(t)$ associated with the $n$ nodes are not in boldface. For a vector $y$, we will also sometimes use $[y]_j$ to denote its $j$’th entry. For a matrix $A$, we will use $A_{ij}$ or $[A]_{ij}$ to denote its $i,j$’th entry. We use $\mathbf{1}$ to denote the vector of ones, and $\|y\|$ for the Euclidean norm of a vector $y$.

II. Problem, Algorithm and Main Result

We consider a network of $n$ nodes which would like to collectively solve the following minimization problem:

$$\minimize F(z) \equiv \sum_{i=1}^{n} f_i(z) \quad \text{over } z \in \mathbb{R}^d,$$

where only node $i$ has any knowledge of the convex function $f_i : \mathbb{R}^d \rightarrow \mathbb{R}$. Moreover, we assume that node $i$ has access to the convex function $f_i : \mathbb{R}^d \rightarrow \mathbb{R}$ only through the ability to generate noisy samples of its subgradient, i.e., given a point $u \in \mathbb{R}^d$ node $i$ can generate

$$g_i(u) = s_i(u) + N_i(u), \quad (1)$$

where $s_i(u)$ is a subgradient of $f_i$ at $u$ and $N_i(u)$ is an independent random vector with zero mean, i.e., $\mathbb{E}[N_i(u)] = 0$. We assume the noise-norm $\|N_i(u)\|$ is almost surely bounded, i.e., each time a noisy subgradient is generated with probability $1$ we have

$$\|N_i(u)\| \leq C_i \|u\| + c_i \quad \text{for all } u \in \mathbb{R}^d \text{ and } i, \quad (2)$$

where $C_i, c_i$ are some scalars. The preceding relation is satisfied for example when each random vector $N_i(u)$ is generated from a distribution with compact support.

We make the assumption that at each time $t$, node $i$ can only send messages to its out-neighbors in some directed graph $G(t)$, where the graph $G(t)$ has vertex set $\{1, \ldots, n\}$ and edge set $E(t)$. We will be assuming that the sequence $\{G(t)\}$ is $B$-strongly-connected, which means that there is a positive integer $B$ such that the graph with edge set

$$E_B(k) = \bigcup_{i=kB}^{(k+1)B-1} E(i)$$

is strongly connected for each $k \geq 0$. Intuitively, we are assuming the time-varying network $G(t)$ must be repeatedly connected over sufficiently long time scales.

We use $N_i^\text{in}(t)$ and $N_i^\text{out}(t)$ denote the in- and out-neighborhoods of node $i$ at time $t$, respectively, where by convention node $i$ is always considered to be an in- and out-neighbor of itself, so $i \in N_i^\text{in}(i)(t)$, $i \in N_i^\text{out}(i)$ for all $i, t$. We use $d_i(t)$ to denote the out-degree of node $i$, and we assume that every node $i$ knows its out-degree $d_i(t)$ at every time $t$.

We will analyze a version of the subgradient-push method of [22], where each node $i$ maintains vector variables $z_i(t), x_i(t), w_i(t) \in \mathbb{R}^d$, as well as a scalar variable $y_i(t)$. These quantities are updated according to the following rules: for all $t \geq 0$ and all $i = 1, \ldots, n$,

$$w_i(t+1) = \sum_{j \in N_i^\text{in}(t)} \frac{x_j(t)}{d_j(t)},$$

$$y_i(t+1) = \sum_{j \in N_i^\text{out}(t)} \frac{y_j(t)}{d_j(t)},$$

$$z_i(t+1) = \frac{w_i(t+1)}{y_i(t+1)},$$

$$x_i(t+1) = w_i(t+1) - \alpha(t+1)g_i(z_i(t+1)), \quad (3)$$

where the variables $y_i(t)$ are initialized as $y_i(0) = 1$ for all $i$. Here, we use $g_i(z_i(t+1))$ to abbreviate the notation $g_i(z_i(t+1))$ (see Eq. (1)). The positive stepsize $\alpha(t+1)$ will be specified later.

These updates have a simple physical implementation: each node $j$ broadcasts the quantities $x_j(t)/d_j(t), y_j(t)/d_j(t)$ to all of the nodes $i$ in its out-neighborhood. Each neighbor $i$ then sums the received messages to obtain $w_i(t+1)$ and $y_i(t+1)$. The updates of $z_i(t+1), x_i(t+1)$ do not require any communications among the nodes at step $t$. For more details on subgradient-push and its motivation we refer the reader to the paper [22].

Our previous work in [22] provided a rate estimate for a suitable averaged version of the variables $z_i(t)$ with the stepsize choice $\alpha(t) = 1/\sqrt{t}$. In particular, we showed in [15] that, for each $i = 1, \ldots, n$, a suitably averaged version of $z_i(t)$ converges to the same global minimum of the function $F(z)$ at a rate of $O((\ln t)/\sqrt{t})$. Our main contribution in this paper is an improved convergence rate estimate $O((\ln t)/t)$ under some stronger assumptions on the functions $f_i$.

Recall that a convex function $f : \mathbb{R}^d \rightarrow \mathbb{R}$ is $\mu$-strongly convex with $\mu > 0$ if the following relation holds for all $x, y \in \mathbb{R}^d$:

$$f(x) - f(y) \geq g'(y)(x-y) + \frac{\mu}{2}\|x-y\|^2,$$

where $g(y)$ is any subgradient of $f(z)$ at $z = y$.

We next provide precise statements of our improved rate estimates. For convenience, we define

$$\bar{x}(t) = \frac{1}{n} \sum_{j=1}^{n} x_j(t)$$
to be the vector which averages all the $x_j(t)$ at each node. Furthermore, let us introduce some notation for the assumptions we will be making.

**Assumption 1:**
(a) The graph sequence $\{G(t)\}$ is $B$-strongly-connected.
(b) Each function $f_i$ is $\mu_i$-strongly convex with $\mu_i > 0$.

Note that Assumption 1(b) implies the existence of a unique global minimizer $z^*$ of $F(z)$.

One of our technical innovations will be to resort to a somewhat unusual type of averaging motivated by the work in [20]. Specifically, we will require each node to work in $\{20\}$. Specifically, we will require each node to require the nodes only to compute the vector which averages all the $x_j(t)$ at each node. Furthermore, let us introduce some notation for the assumptions we will be making.

The assumptions we will be making.

This can easily be done recursively, e.g., by setting $\hat{z}_i(t+1) = t \hat{z}_i(t) + S(t) \hat{z}_i(t)$ for $t \geq 1$,

$$
\hat{z}_i(t+1) = \frac{t \hat{z}_i(t) + S(t) \hat{z}_i(t)}{S(t+1)} \quad \text{for} \quad t \geq 1, \quad (5)
$$

where $S(t) = (t-1)/2$ for $t \geq 2$.

We are now ready to state our first main result, which deals with the speed at which the averaged iterates $\hat{z}_i(t)$ we have just described converge to the global minimizer $z^*$ of $F(z)$.

**Theorem 1:** Suppose Assumption 1 is satisfied and $\alpha(t) = \frac{\ell}{t}$ for $t \geq 1$, where the constant $p$ is such that

$$
p \sum_{i=1}^{n} \lambda_i = 4.
$$

(6)

Suppose further that there exists a scalar $D$ such that with probability 1, $\sup_i ||z_i(t)|| \leq D$. Then, we have for all $i = 1, \ldots, n$,

$$
E \left[ F(\hat{z}_i(\tau)) - F(z^*) + \sum_{j=1}^{n} \mu_j ||\hat{z}_j(\tau) - z^*||^2 \right] 
\leq \frac{40L}{\tau \delta} \frac{\lambda}{1-\lambda} \sum_{j=1}^{n} ||x_j(0)||_1 
+ \frac{32pn}{\tau \delta} \frac{\sum_{j=1}^{n} L_j B_j}{1-\lambda} (1 + \ln(\tau - 1)) 
+ \frac{8pn}{\tau \delta} \frac{LB_i}{1-\lambda} (1 + \ln(\tau - 1)) + \frac{p}{\tau} \sum_{j=1}^{n} ||B_j||^2,
$$

where $L_i$ is the largest-possible 1-norm of any subgradient of $f_i$ on the ball of radius $nD$ around the origin, $B_i = L_i + C_iD + c_i$, while the scalars $\lambda \in (0, 1)$ and $\delta > 0$ are functions of the graph sequence $\{G(t)\}$ which satisfy

$$
\delta \geq \frac{1}{nB}, \quad \lambda \leq \left( 1 - \frac{1}{nB} \right)^{1/(nB)}.
$$

Moreover, if each of the graphs $G(t)$ is regular then $\delta = 1$, $\lambda \leq \min \left\{ \left( 1 - \frac{1}{4n^2} \right)^{1/B}, \max_{t \geq 0} \sigma_2(A(t)) \right\}$, where $A(t)$ is defined by

$$
A_{ij}(t) = \begin{cases} 1/d_j(t) & \text{whenever} \ j \in N_i(t), \\ 0 & \text{otherwise}. \end{cases}
$$

(7)

and $\sigma_2(A)$ is the second-largest singular value of $A$.

Note that each term on the right-hand side of the bound in the above theorem has a $\tau$ in the denominator and a $\ln(\tau - 1)$ or a constant in the numerator. The convergence time above should therefore be interpreted as proving a decay with time which decreases at an expected $O((\ln t)/t)$ rate with the number of iterations $t$. Note that this is an extended and corrected version of a result from the conference version of this paper [21].

We note that the bounds for the constants $\delta$ and $\lambda$ which appear in the bound are rather large in the most general case; in particular, they grow exponentially in the number of nodes $n$. At present, this scaling appears unavoidable: those constants reflect the best available bounds on the performance of average consensus protocols in directed graphs, and it is an open question whether average consensus on directed graphs can be done in time polynomial in $n$. In the case of regular graphs, the bounds scale polynomially in $n$ due the availability of good bounds on the convergence of consensus. Our results therefore further motivate problem of finding consensus algorithms with good convergence times.

Finally, we remark that choosing a stepsize parameter $p$ so that Eq. (6) is satisfied is most easily done by instead insuring that $p(\min_i \mu_i)/n > 4$. This is a more conservative condition than that of Eq. (6) but ensuring it requires the nodes only to compute $\min_i \mu_i$. This is more convenient because the minimum of any collection of numbers $r_1, \ldots, r_n$ (with $r_i$ stored at node $i$) can be easily computed by the following distributed protocol: node $i$ sets its initial value to $r_i$ and then repeatedly replaces its value with the minimum of the values of its in-neighbors. It is easy to see that on any fixed network, this process converges to the minimum in as many steps as the diameter. Furthermore, on any $B$-strongly-connected sequence this processes converges in the optimal $O(nB)$ steps. Thus, the pre-processing required to come up with a suitable step-size parameter $p$ is reasonably small.

A shortcoming of Theorem 1 is that we must assume that the iterates $z_i(t)$ remain bounded (as opposed to

\footnote{A directed graph $G(t)$ is regular if every out-degree and every in-degree of a node in $G(t)$ equals $d(t)$ for some $d(t)$.}
obtaining this as a by-product of the theorem). This is a common situation in the analysis of subgradient-type methods in non-differentiable optimization: the boundedness of the iterates or their subgradients often needs to be assumed in advance in order to obtain a result about convergence rate.

We next show that we can remedy this shortcoming at the cost of imposing additional assumptions on the functions \( f_i \), namely that they are differentiable and their gradients are Lipschitz.

**Assumption 2:** Each \( f_i \) is differentiable and its gradients are Lipschitz continuous, i.e., for a scalar \( M_i > 0 \),
\[
\| \nabla f_i(x) - \nabla f_i(y) \| \leq M_i \| x - y \| \quad \text{for all } x, y \in \mathbb{R}^d.
\]

**Theorem 2:** Suppose that Assumption 1 and Assumption 2 hold and suppose \( \lim_{t \to \infty} \alpha(t) = 0 \). Then, there exists a scalar \( D \) such that with probability 1, \( \sup_t \| z_i(t) \| \leq D \) for all \( i \).

The proof of this theorem is constructive in the sense than an explicit expression for \( D \) can be derived in terms of \( \alpha(t) \) and the growth of level sets of the functions \( f_j \). Putting Theorems 1 and 2 together, we obtain our main result: for strongly convex functions \( f \), the (sub)gradient-push with appropriately chosen step-size and averaging strategy converges at a \( O((\ln t)/t) \) rate.

**III. PROOF OF THEOREM [1]**

We briefly sketch the main ideas of the proof of Theorem [1]. First, we will argue that if the subgradient terms in the subgradient-push protocol are bounded, then as a consequence of the decaying stepsize \( \alpha(t) \), the protocol will achieve consensus. We will then analyze the evolution of the average \( \bar{x}(t) \) and show that, as a consequence of the protocol achieving consensus, \( \bar{x}(t) \) satisfies approximately the same recursion as the iterates of the ordinary subgradient method. Finally, we rely on a recent idea from [22] to show that, for a noisy gradient update on a strongly convex function, a decay of nearly \( 1/t \) can be achieved by a simple averaging of iterates that places more weight on recent iterations.

Our starting point is an analysis of a perturbation of the so-called push-sum protocol of [13] for computing averages in directed networks. We next describe this perturbed push-sum method. Every node \( i \) maintains scalar variables \( x_i(t), y_i(t), z_i(t), w_i(t) \), where \( y_i(0) = 1 \) for all \( i \). Every node \( i \) updates these variables according to the following rule: for \( t \geq 0 \),
\[
w_i(t + 1) = \sum_{j \in N_i(t)} \frac{x_j(t)}{d_j(t)},
\]
\[
y_i(t + 1) = \sum_{j \in N_i(t)} \frac{y_j(t)}{d_j(t)},
\]
\[
z_i(t + 1) = w_i(t + 1),
\]
\[
x_i(t + 1) = w_i(t + 1) + \epsilon_i(t + 1),
\]
where \( \epsilon_i(t) \) is some (perhaps adversarially chosen) perturbation at time \( t \). Without the perturbation term \( \epsilon_i(t) \), the method in Eq. (8) is called **push-sum**. For the perturbed push-sum method above in Eq. (8), we have that the following is true.

**Lemma 1 ([22]):** Consider the sequences \( \{z_i(t)\} \), \( i = 1, \ldots, n \), generated by the method in Eq. (8). Assuming that the graph sequence \( \{G(t)\} \) is \( B \)-strongly-connected, we have that for all \( t \geq 1 \),
\[
\left| z_i(t + 1) - \frac{1}{n} x(t) \right| \leq \frac{8}{\delta} \left( \lambda\|x(0)\|_1 + \sum_{s=1}^{t} \lambda^{t-s} \|e(s)\|_1 \right),
\]
where \( e(s) \) is a vector in \( \mathbb{R}^n \) which stacks up the scalar variables \( e_i(s) \), \( i = 1, \ldots, n \), and \( \delta, \lambda \) satisfy the same inequalities as in Theorem [1].

We refer the reader to [22] for a proof where this statement is Lemma 1.

**Corollary 1:** Consider the update of Eq. (8) with the scalar variables \( x_i(t), y_i(t), z_i(t), \epsilon_i(t) \) replaced by the vector variables \( x_i(t), w_i(t) \) for each \( i = 1, \ldots, n \). Assuming that the graph sequence \( \{G(t)\} \) is \( B \)-strongly-connected, for all \( i = 1, \ldots, n, t \geq 1 \) we have
\[
\left\| z_i(t + 1) - \frac{1}{n} \sum_{j=1}^{n} x_j(t) \right\| \leq \frac{8}{\delta} \left( \lambda \sum_{i=1}^{n} \|x_i(0)\|_1 + \sum_{s=1}^{t} \lambda^{t-s} \sum_{i=1}^{n} \|e_i(s)\|_1 \right),
\]
where \( \delta, \lambda \) satisfy the same inequalities as in Theorem [1].

**Corollary 2:** Under the assumptions of Corollary 1 and assuming that the perturbation vectors \( e_i(t) \) are random vectors satisfying for some scalar \( D > 0 \),
\[
\mathbb{E}[\|e_i(t)\|_1] \leq \frac{D}{t} \quad \text{for all } i = 1, \ldots, n \text{ and all } t \geq 1,
\]
we have that for all \( i = 1, \ldots, n \) and all \( \tau \geq 1 \),
Now, let $v \in \mathbb{R}^d$ be an arbitrary vector. From relation \([9]\) we can see that for all $t \geq 0$,
\[
\|\bar{x}(t+1) - v\|^2 \leq \|\bar{x}(t) - v\|^2
\]
\[
- \frac{2\alpha(t+1)}{n} \sum_{j=1}^{n} g_j(t+1)(\bar{x}(t) - v)
\]
\[
+ \frac{\alpha^2(t+1)}{n^2} \sum_{j=1}^{n} \|g_j(t+1)\|^2.
\]
Taking expectations of both sides with respect to $\mathcal{F}_t$, and using $g_j(t+1) = \nabla f_j(z_j(t+1)) + N_j(z_j(t+1))$ (see Eq. \([12]\)) and the relation
\[
\mathbb{E}[N_j(z_j(t+1)) \mid \mathcal{F}_t] = \mathbb{E}[N_j(z_j(t+1)) \mid z_j(t+1)] = 0,
\]
we obtain
\[
\mathbb{E}\left[\|\bar{x}(t+1) - v\|^2 \mid \mathcal{F}_t\right] \leq \mathbb{E}\left[\|\bar{x}(t) - v\|^2\right]
\]
\[
- \frac{2\alpha(t+1)}{n} \sum_{j=1}^{n} \nabla f_j'(z_j(t+1))(\bar{x}(t) - v)
\]
\[
+ \frac{\alpha^2(t+1)}{n^2} \sum_{j=1}^{n} \|g_j(t+1)\|^2.
\]
Next, we upper-bound the last term in the preceding relation. By using the inequality $\left(\sum_{j=1}^{n} a_j\right)^2 \leq n \sum_{j=1}^{n} a_j^2$ and Theorem \([3]\) and noting that the bounds of Theorem \([3]\) also hold for the Euclidean norm (since $\|x\| \leq \|x\|_1$ for all $x$), we obtain
\[
\|\sum_{j=1}^{n} g_j(t+1)\|^2 \leq n \sum_{j=1}^{n} \|g_j(t+1)\|^2 \leq n B_j^2,
\]
where $B_j$ are constants from Theorem \([3]\).

\textbf{Proof:} Note that the matrix $A(t)$ defined in the statement of Theorem 1 (see Eq. \([7]\)) is column stochastic, so that $1'u = 1'A(t)u$ for any vector $u \in \mathbb{R}^n$. Consequently, for the stochastic gradient-push update of Eq. \([3]\) we have
\[
\bar{x}(t+1) = \bar{x}(t) - \frac{\alpha(t+1)}{n} \sum_{j=1}^{n} g_j(t+1).
\]

Note that as a consequence of Eq. \([3]\), we have that $\|\bar{x}(t)\| \leq n \max_j \|z_j(t)\|$. By definition of $L_j$, we have that all subgradients of $f_j$ at $z_j(t)$ and $\bar{x}(t)$ are upper bounded by $L_j$ in the 1-norm. Thus, using the Cauchy-Schwarz inequality,
\[
\nabla f_j'(z_j(t+1))(\bar{x}(t) - z_j(t+1)) \geq -L_j \|\bar{x}(t) - z_j(t+1)\|.
\]
As for the term $\nabla f_j'(z_j(t+1))(z_j(t+1) - v)$, we use the fact that the function $f_j$ is $\mu_j$-strongly convex to obtain

$$\nabla f_j'(z_j(t+1))(z_j(t+1) - v) \geq f_j(z_j(t+1)) - f_j(v) + \frac{\mu_j}{2} \|z_j(t+1) - v\|^2.$$  

By writing $f_j(z_j(t+1)) - f_j(v) = (f_j(z_j(t+1)) - f_j(\bar{x}(t))) + (f_j(\bar{x}(t)) - f_j(v))$ and using the Lipschitz continuity of $f_j$ (implied by the subgradient boundedness), we further obtain

$$\nabla f_j'(z_j(t+1))(z_j(t+1) - v) \geq -L_j \|z_j(t+1) - \bar{x}(t)\| + f_j(\bar{x}(t)) - f_i(v) + \frac{\mu_j}{2} \|z_j(t+1) - v\|^2.$$  

By substituting the estimates of Eqs. (12)–(13) back in relation (11), and using $F(x) = \sum_{j=1}^n f_j(x)$ we obtain

$$\sum_{i=1}^n \nabla f_j'(z_j(t+1))(\bar{x}(t) - v) \geq F(\bar{x}(t)) - F(v) + \frac{1}{2} \sum_{j=1}^n \mu_j \|z_j(t+1) - v\|^2 - 2 \sum_{j=1}^n L_j \|z_j(t+1) - \bar{x}(t)\|.$$  

Plugging this relation into Eq. (10), we obtain the statement of this lemma. \[\square\]

With Lemma 2 in place, we are now ready to provide the proof of Theorem 1. Besides Lemma 2 our arguments will also crucially rely on the results established earlier for the perturbed push-sum method.

**Proof of Theorem 1** The function $F = \sum_{i=1}^n f_i$ has a unique minimum which we will denote by $z^*$. In Lemma 2 we let $v = z^*$ to obtain for all $t \geq 0$,

$$\mathbb{E}[\|\bar{x}(t+1) - z^*\|^2 | F_t] \leq \|\bar{x}(t) - z^*\|^2 - 2\alpha(t+1) \frac{p}{n} (F(\bar{x}(t)) - F(z^*)) - \frac{\alpha(t+1)}{n} \sum_{j=1}^n \mu_j \|z_j(t+1) - z^*\|^2 + \frac{4\alpha(t+1)}{n} \sum_{j=1}^n L_j \|z_j(t+1) - \bar{x}(t)\| + \frac{\alpha^2(t+1)}{n} \sum_{j=1}^n B_j^2.$$  

Next, we estimate the term $F(\bar{x}(t)) - F(z^*)$ in the above equation by breaking it into two parts. On the one hand, $F(\bar{x}(t)) - F(z^*) \geq \frac{1}{2} \sum_{j=1}^n \mu_j \|\bar{x}(t) - z^*\|^2$. On the other hand, since the function $F$ is Lipschitz continuous with constant $L = L_1 + \cdots + L_n$ we also have that for any $i = 1, \ldots, n$,

$$F(\bar{x}(t)) - F(z^*) = (F(\bar{x}(t)) - F(z_i(t+1))) + (F(z_i(t+1)) - F(z^*)) \geq -L \|z_i(t+1) - \bar{x}(t)\| + F(z_i(t+1)) - F(z^*).$$

Therefore, using the preceding two estimates we obtain for all $i = 1, \ldots, n$,

$$2(F(\bar{x}(t)) - F(z^*)) \geq \frac{1}{2} \sum_{j=1}^n \mu_j \|\bar{x}(t) - z^*\|^2 - L \|z_i(t+1) - \bar{x}(t)\| + (F(z_i(t+1)) - F(z^*)) \geq \frac{1}{2} \sum_{j=1}^n \mu_j \|\bar{x}(t) - z^*\|^2 - L \|z_i(t+1) - \bar{x}(t)\| + (F(z_i(t+1)) - F(z^*)).$$

Combining relation (15) with Eq. (14) and simultaneously plugging in the expression for $\alpha(t)$, we see that for all $i = 1, \ldots, n$ and all $t \geq 0$,

$$\mathbb{E}[\|\bar{x}(t+1) - z^*\|^2 | F_t] \leq \left(1 - \frac{2}{t+1}\right) \|\bar{x}(t) - z^*\|^2 - \frac{p}{n(t+1)} (F(z_i(t+1)) - F(z^*)) + \frac{pL}{n(t+1)} \|z_i(t+1) - \bar{x}(t)\| - \frac{p}{n(t+1)} \sum_{j=1}^n \mu_j \|z_j(t+1) - z^*\|^2 + \frac{4p}{n(t+1)} \sum_{j=1}^n L_j \|z_j(t+1) - \bar{x}(t)\| + \frac{p^2}{(t+1)^2} \frac{q^2}{n},$$

where $p^2 = \sum_{j=1}^n B_j^2$. We multiply the preceding relation by $t(t+1)$ and obtain that for all $i = 1, \ldots, n$ and all $t \geq 1$,

$$(t+1)t\mathbb{E}[\|\bar{x}(t+1) - z^*\|^2 | F_t] \leq t(t-1) \|\bar{x}(t) - z^*\|^2 - \frac{pt}{n} (F(z_i(t+1)) - F(z^*)) + \frac{pLt}{n} \|z_i(t+1) - \bar{x}(t)\| - \frac{pt}{n} \sum_{j=1}^n \mu_j \|z_j(t+1) - z^*\|^2 + \frac{4pt}{n} \sum_{j=1}^n L_j \|z_j(t+1) - \bar{x}(t)\| + \frac{p^2t}{t+1} \frac{q^2}{n}.$$  

Taking expectations and applying the above identity...
reversely, we obtain that all \( \tau \geq 2 \),
\[
\tau(\tau - 1)\mathbb{E}[\| \mathbf{x}(\tau) - \mathbf{z}^* \|^2] \leq \\
\frac{p}{n} \sum_{t=1}^{\tau-1} t \mathbb{E} \left[ F(\mathbf{z}(t+1)) - F(\mathbf{z}^*) + \sum_{j=1}^{n} \mu_j \| \mathbf{z}_j(t+1) - \mathbf{z}_j^* \|^2 \right] \\
+ \frac{4p}{n} \sum_{t=1}^{\tau-1} L_j \mathbb{E}[\| \mathbf{z}_j(t+1) - \mathbf{x}(t)\|] \\
+ \frac{pL}{n} \sum_{t=1}^{\tau-1} t \mathbb{E}[\| \mathbf{z}_j(t+1) - \mathbf{x}(t)\|] + \frac{p^2q}{n} \sum_{t=1}^{\tau-1} \frac{t}{t+1},
\]
(17)

By viewing the stochastic gradient-push method as an instance of the perturbed push-sum protocol, we can apply Corollary 2 with \( e_i(t) = \alpha(t) g_i(t) \). Since \( \|g_i(t)\|_1 \leq B_i \) for all \( i, t \) (by Theorem 2) and since \( \alpha(t) \leq p \) for all \( t \) we see that
\[
\mathbb{E}[\|e_i(t)\|_1] \leq pB_i \quad \text{for all } i \text{ and } t \geq 1.
\]
Thus, by Corollary 2 we obtain for all \( i = 1, \ldots, n \),
\[
\mathbb{E}[\sum_{t=1}^{\tau-1} \| \mathbf{z}_i(t+1) - \frac{\sum_{j=1}^{n} \mathbf{x}_j(t)}{n} \|] \leq \frac{8}{\delta^2} \frac{\lambda}{1 - \lambda} \sum_{j=1}^{n} \| \mathbf{x}_j(0) \|_1 \\
+ \frac{8}{\delta} \frac{mnB_i}{1 - \lambda} (1 + \ln(\tau - 1)).
\]
(18)

By substituting (18) into inequality (17), dividing both sides by \( (\tau - 1) \), and rearranging, we obtain that for all \( \tau \geq 2 \),
\[
\frac{p}{n\tau(\tau - 1)} \sum_{t=1}^{\tau-1} t \mathbb{E} \left[ F(\mathbf{z}(t+1)) - F(\mathbf{z}^*) + \sum_{j=1}^{n} \mu_j \| \mathbf{z}_j(t+1) - \mathbf{z}_j^* \|^2 \right] \\
\leq \frac{5p}{n\tau^2} \frac{8}{\delta^2} \frac{\lambda}{1 - \lambda} \sum_{j=1}^{n} \| \mathbf{x}_j(0) \|_1 \\
+ \frac{4p}{n\tau^2} \frac{8}{\delta} \frac{mnL_jB_j}{1 - \lambda} (1 + \ln(\tau - 1)) \\
+ \frac{pL}{n\tau^2} \frac{8}{\delta} \frac{mnB_i}{1 - \lambda} (1 + \ln(\tau - 1)) + \frac{p^2q}{\tau n}
\]
(19)

By convexity we have
\[
\sum_{t=1}^{\tau-1} t \left( F(\mathbf{z}(t+1)) - F(\mathbf{z}^*) + \sum_{j=1}^{n} \mu_j \| \mathbf{z}_j(t+1) - \mathbf{z}_j^* \|^2 \right) \\
\geq F(\mathbf{z}(\tau)) - F(\mathbf{z}^*) + \sum_{j=1}^{n} \mu_j \| \mathbf{z}_j(\tau) - \mathbf{z}_j^* \|^2
\]
(20)

Now, Eqs. (19) and (20) conclude the proof.

IV. PROOF OF THEOREM 2

We begin by briefly sketching the main idea of the proof. The proof proceeds by simply arguing that if \( \max_i \| \mathbf{z}_i(t) \| \) get large, it decreases. Since the stochastic subgradient-push protocol (Eq. 3) is somewhat involved, proving this will require some involved arguments relying on the level-set boundedness of strongly convex functions with Lipschitz gradients and some special properties of element-wise ratios of products of column-stochastic matrices.

Our starting point is a lemma that exploits the structure of strongly convex functions with Lipschitz gradients.

**Lemma 3:** Let \( q : \mathbb{R}^d \to \mathbb{R} \) be a \( \mu \)-strongly convex function with \( \mu > 0 \) and have Lipschitz continuous gradients with constant \( M > 0 \). Let \( v \in \mathbb{R}^d \) and let \( u \in \mathbb{R}^d \) be defined by
\[
u = v - \alpha (\nabla q(v) + \phi(v)),
\]
where \( \phi : \mathbb{R}^d \to \mathbb{R}^d \) is a mapping such that for \( C, c > 0 \),
\[
\|\phi(v)\| \leq C\|v\| + c \quad \text{for all } v \in \mathbb{R}^d,
\]
and \( \alpha \in (0, \frac{\mu}{(M + cC)^2}] \). Then, there exists a compact set \( \mathcal{V} \subset \mathbb{R}^d \) such that
\[
\|u\| \leq \left\{ \begin{array}{ll}
\|v\| & \text{for all } v \notin \mathcal{V} \\
R & \text{for all } v \in \mathcal{V}
\end{array} \right.
\]
with \( R = \gamma(1 + C) \max_{z \in \mathcal{V}} \{\|z\| + \|\nabla q(z)\|\} + c \gamma \) where \( \gamma = \max\{1, \frac{\mu}{(M + cC)^2}\} \).

**Proof:** For the vector \( u \) we have
\[
\|u\|^2 = \|v\|^2 - 2\alpha q(v) + \alpha^2 \|\nabla q(v) + \phi(v)\|^2 \\
\leq (1 - \alpha \mu)\|v\|^2 - 2\alpha (q(v) - q(0)) \\
+ \alpha^2 \|\nabla q(v) + \phi(v)\|^2,
\]
where the inequality follows by the strong convexity of the function \( q \). For the last term in the preceding relation, we write
\[
\|\nabla q(v) + \phi(v)\|^2 \leq 2\|\nabla q(v)\|^2 + 2\|\phi(v)\|^2,
\]
(21)
where we use the following inequality
\[
(a + b)^2 \leq 2(a^2 + b^2) \quad \text{for any } a, b \in \mathbb{R}.
\]
(22)

We can further write
\[
\|\nabla q(v)\|^2 \leq (\|\nabla q(v) - \nabla q(0)\| + \|\nabla q(0)\|)^2 \\
\leq 2M^2\|v\|^2 + 2\|\nabla q(0)\|^2,
\]
(23)
where the last inequality is obtained by using Eq. (22) and by exploiting the Lipschitz gradient property of \( q \). Similarly, using the given growth-property of \( \|\phi(v)\| \) and relation in (22) we obtain
\[
\|\phi(v)\|^2 \leq 2C^2\|v\|^2 + 2c^2.
\]
(24)
By substituting Eqs. (23)–(24) in relation (21), we find
\[
\|\nabla q(v) + \phi(v)\|^2 \leq 4(M^2 + C^2)\|v\|^2 + 4\|\nabla q(0)\|^2 + 4c^2.
\]
Therefore, we obtain
\[
\|u\|^2 \leq (1 - \alpha(\mu - 4\alpha(M^2 + C^2)))\|v\|^2 \\
- 2\alpha(q(v) - q(0)) + 4\alpha^2(\|\nabla q(0)\|^2 + c^2),
\]
which for \( \alpha \in (0, \frac{\mu}{4(M^2 + C_1^2)}) \) yields
\[
\|u\|^2 \leq \|v\|^2 - 2\alpha \left( q(v) - q(0) - 2\alpha(\|\nabla q(0)\|^2 + c^2) \right).
\]

If \( v \) is such that \( q(v) \geq q(0) + 2\alpha(\|\nabla q(0)\|^2 + c^2) \), then we obtain \( \|u\|^2 \leq \|v\|^2 \).

Define the set \( \mathcal{V} \) to be the level set of \( q \), as follows:
\[
\mathcal{V} = \{ z \mid q(z) \leq q(0) + 2\alpha(\|\nabla q(0)\|^2 + c^2) \}.
\]

Being the level-set of a strongly-convex function, the set \( \mathcal{V} \) is compact [2] (see Proposition 2.3.1(b), page 93).

Lemma 4: Suppose \( P \) is an \( n \times n \) column-stochastic matrix with positive diagonal entries, and let \( u, v \in \mathbb{R}^n \) with the vector \( v \) having all entries positive. Consider the vectors \( \hat{u} \) and \( \hat{v} \) given, respectively, by
\[
\hat{u} = Pu, \quad \hat{v} = Pv.
\]

Define the vectors \( r \) and \( \hat{r} \) with their \( i \)'th entries given by \( r_i = u_i/v_i \) and \( \hat{r}_i = \hat{u}_i/\hat{v}_i \), respectively. Then, we have
\[
\hat{r} = Qr,
\]
where \( Q \) is a row-stochastic matrix.

Proof: Indeed, note that
\[
\hat{u}_i = \sum_{j=1}^{n} P_{ij}u_j \quad \text{for all } i.
\]

Since \( \hat{u}_i = \hat{v}_i \hat{u}_i \) and \( u_j = v_j r_j \), the preceding equation can be rewritten as
\[
\hat{v}_i \hat{r}_i = \sum_{j=1}^{n} P_{ij}v_j r_j.
\]

Since \( v \) has all entries positive and \( P \) has positive diagonal entries, it follows that \( \hat{v} \) also has all entries positive. Therefore,
\[
\hat{r}_i = \frac{1}{v_i} \sum_{j=1}^{n} P_{ij}v_j r_j = \sum_{j=1}^{n} \frac{P_{ij}v_j}{v_i} r_j.
\]

Define the matrix \( Q \) from this equation, i.e., \( Q_{ij} = \frac{P_{ij}v_j}{v_i} \) for all \( i,j \). The fact that \( Q \) is row-stochastic follows from \( \hat{v} = Pv \).

With this lemma in place, we now prove the theorem.

Proof of Theorem 2: Letting \( y(t) \) be the vector with entries \( y_i(t) \), we can write \( y(t + 1) = A(t)y(t) \), where \( A(t) \) is the matrix given in Eq. (7). Thus, since \( y_1(0) = 1 \) for all \( i \), we have
\[
y(t) = A(t)A(t-1) \cdots A(0)1 \quad \text{for all } i \text{ and } t \geq 1,
\]
where \( 1 \) is the vector with all entries equal to 1. Under Assumption [4], we have shown in [22] (see there Corollary 2(b)) that for all \( i \),
\[
\delta = \inf_{t=0,1,\ldots} \left( \min_{1 \leq i \leq n} [A(t)A(t-1) \cdots A(0)1]_i \right) > 0.
\]

Therefore, we have
\[
y_i(t) \geq \delta \quad \text{for all } i \text{ and } t. \tag{25}
\]

Thus, using the definition of \( x_i(t+1) \), we can see that
\[
x_i(t) = y_i(t) \left( z_i(t) - \frac{\alpha(t)}{y_i(t)} g_i(t) \right) \quad \text{for all } t \geq 1,
\]

implying that for all \( i \) and \( t \geq 1 \),
\[
\frac{x_i(t)}{y_i(t)} = \frac{z_i(t) - \alpha(t)y_i(t)}{y_i(t)} g_i(t). \tag{26}
\]

Since the matrix \( A(t)A(t-1) \cdots A(0) \) is column stochastic and \( y(0) = 1 \), we have that \( \sum_{i=1}^{n} y_i(t) = n \).

Therefore, \( y_i(t) \leq n \), which together with Eq. (25) and \( \alpha(t) \to 0 \) yields
\[
\lim_{t \to \infty} \frac{\alpha(t)}{y_i(t)} = 0 \quad \text{for all } i.
\]

Therefore, for every \( i \), there is a time \( \tau_i > 1 \) such that \( \alpha(t)/y_i(t) \leq \frac{\mu}{4(M^2 + C_1^2)} \) for all \( t \geq \tau_i \). Hence, for each \( i \), Lemma 3 applies to the vector \( x_i(t)/y_i(t) \) for \( t \geq \tau_i \).

By Lemma 3, it follows that for each function \( f_i \), there is a compact set \( \mathcal{Y}_i \) and a time \( \tau_i \) such that for all \( t \geq \tau_i \),
\[
\frac{\|x_i(t)\|}{y_i(t)} \leq \begin{cases} \|z_i(t)\| & \text{if } z_i(t) \notin \mathcal{Y}_i, \\ R_i & \text{if } z_i(t) \in \mathcal{Y}_i. \end{cases} \tag{27}
\]

Let \( \tau = \max_i \tau_i \). By using the mathematical induction on the time, we will prove that
\[
\max_{1 \leq i \leq n} \|z_i(t)\| \leq \bar{R} \quad \text{for all } t \geq \tau, \tag{28}
\]
where \( \bar{R} = \max \{ \max_i R_i, \max_j \|z_j(\tau)\| \} \). Evidently, relation (28) is true for \( t = \tau \). Suppose it is true for some \( t > \tau \). Then, at the time \( t \), by Eq. (27) we have
\[
\frac{\|x_i(t)\|}{y_i(t)} \leq \max_i \left( R_i, \max_j \|z_j(t)\| \right) \leq \bar{R} \quad \text{for all } i, \tag{29}
\]
where the last inequality follows by the induction hypothesis. Next, we use Lemma 3 with \( v = y(t) \), \( P = A(t) \), and \( u \) taken as the vector of the \( \ell \)'th coordinates of the vectors \( x_j(t) \), \( j = 1, \ldots, n \), where the coordinate index \( \ell \) is arbitrary. In this way, we obtain that each
vector $z_i(t + 1)$ is a convex combination of the vectors $x_i(t)/y_j(t)$, i.e.,

$$z_i(t + 1) = \sum_{j=1}^{n} Q_{ij}(t) \frac{x_j(t)}{y_j(t)}$$

for all $i$ and $t \geq 0$, (30)

where $Q(t)$ is a row stochastic matrix with entries $Q_{ij}(t) = A_{ij}(t)y_j(t)$. By the convexity of the Euclidean norm, it follows that for all $i$,

$$\|z_i(t + 1)\| \leq \sum_{j=1}^{n} Q_{ij}(t) \frac{\|x_j(t)\|}{y_j(t)} \leq \max_{1 \leq j \leq n} \frac{\|x_j(t)\|}{y_j(t)},$$

which together with Eq. (29) yields $\|z_i(t + 1)\| \leq \bar{R}$, thus implying that at time $t + 1$ we have

$$\max_{1 \leq i \leq n} \|z_i(t + 1)\| \leq \bar{R}.$$

Hence, Eq. (28) is valid for all $t \geq \tau$.

Note that the constant $\bar{R}$ is random as it depends on the random vectors $x_i(t), i = 1, \ldots, n$, where the time $\tau$ is deterministic. Then, using Eqs. (26) and (30), we can see that for all $t \geq 1$,

$$\max_{1 \leq i \leq n} \|z_i(t + 1)\| \leq \max_{1 \leq i \leq n} \left( \|z_i(t)\| + \frac{\alpha(t)}{y_j(t)} \|g_j(t)\| \right).$$

In view of noisy gradient relation Eq. (1), the growth-property of the noise (cf. Eq. (2)), the gradient Lipschitz property, and the lower bound on $y_j(t)$ in Eq. (25), it follows that for all $t \geq 1$,

$$\max_{1 \leq i \leq n} \|z_i(t + 1)\| \leq \gamma_1 \max_{1 \leq i \leq n} \|z_i(t)\| + \gamma_2,$$

where $\gamma_1 = 1 + \bar{\alpha}/\delta \max_{j}(M_j + C_j)$, $\gamma_2 = \bar{\alpha}/\delta \max_{j} e_j$, and $\bar{\alpha} = \max_{i} \alpha(t)$. Thus, using the preceding relation recursively for $t = 1, \ldots, \tau - 1$ and the fact that the initial points $x_i(0)$ are deterministic, we conclude that there exists a uniform deterministic bound on $\|z_i(t)\|$ for all $t \geq 1$ and $i$.

V. SIMULATIONS

We report some simulations of the subgradient-push method which experimentally demonstrate its performance. We consider the scalar function $F(\theta) = \sum_{i=1}^{n} p_i(\theta - u_i)^2$ where $u_i$ is a variable that is known only to node $i$. This is a canonical problem in distributed estimation, whereby the nodes are attempting to measure a parameter $\theta$. Each node $i$ measures $u_i = \tilde{\theta} + w_i$, where $w_i$ are jointly Gaussian and zero mean. Letting $p_i$ be the inverse of the variance of $w_i$, the maximum likelihood estimate is the minimizer $\theta^*$ of $F(\theta)$ ($\theta^*$ is unique provided that $p_i > 0$ for at least one $i$). Each $p_i$ is a uniformly random variable taking values between 0 and 1. The initial points $x_i(0)$ are generated as independent random variables, each with a standard Gaussian distribution. This setup is especially attractive since the optimal solution can be computed explicitly (it is a weighted average of the $u_i$) allowing us to see exactly how far from optimality our protocol is at every stage.

The subgradient-push method is run for 200 iterations with the stepsize $\alpha(t) = p/t$ and $p = 2n/(\sum_{i=1}^{n} p_i)$. The graph sequence is constructed over 1000 nodes with a random connectivity pattern.

Figure 1 shows the results obtained for simple random graphs where every node has two out-neighbors, one belonging to a fixed cycle and the other one chosen uniformly at random at each step. The top plot shows how $\ln(\|\tilde{z}_i(t) - \theta^*\|)$ decays on average (over 25 Monte Carlo simulations) for five randomly selected nodes. The bottom plot shows a sample of $\ln(\|\tilde{z}_i(t) - \theta^*\|)$ for a single Monte Carlo run and the same selection of 5 nodes.

Figure 2 illustrates the same quantities for the sequence of graphs which alternate between two (undirected) star graphs.

---

**Fig. 1.** Top plot: the number of iterations ($x$-axis) and the average of $\ln(\|\tilde{z}_i(t) - \theta^*\|)$ ($y$-axis) over 25 Monte Carlo runs for 5 randomly chosen nodes. Bottom plot: a sample of one a single run for the same node selection.

We see that the error decays at a fairly speedy rate, especially given both the relatively large number of
nodes in the system (a thousand) and the sparsity of the graph at each stage (every node has two out-neighbors). Our simulation results suggest the gradient-push methods we have proposed have the potential to be effective tools for network optimization problems. For example, the simulation of Figure 1 shows that a relatively fast convergence time can be obtained if each node can support only a single long-distance out-link.

VI. Conclusion

We have considered a variant of the subgradient-push method of our prior work [22], where the nodes have access to noisy subgradients of their individual objective functions $f_i$. Our main result was that the functions $f_i$ are strongly convex functions with Lipschitz gradients, we have established $O(\ln t/t)$ convergence rate of the method, which is an improvement of the previously known rate $O(\ln t/\sqrt{t})$ for (noiseless) subgradient-push method shown in [22].

Our work suggests a number of open questions. Our bounds on the performance of the (sub)gradient-push directly involve the convergence speed $\lambda$ of consensus on directed graphs. Thus the problem of designing well-performing consensus algorithms is further motivated by this work. In particular, a directed average consensus algorithm with polynomial scaling with $n$ on arbitrary time-varying graphs would lead to polynomial convergence-time scalings for distributed optimization over time-varying directed graphs. However, such an algorithm is not available to the best of the authors’ knowledge.

Moreover, it would be interesting to relate the convergence speed of distributed optimization procedures to the properties possessed by the individual functions. We have begun on this research program here by showing an improved rate for strongly convex functions with Lipschitz gradients. However, one might expect that stronger results might be available under additional assumptions. It is not clear, for example, under what conditions a geometric rate can be achieved when graphs are directed and time-varying, if at all.

REFERENCES

[1] F. Benezit, V. Blondel, P. Thiran, J. Tsitsiklis, and M. Vetterli. Weighted gossip: distributed averaging using non-doubly stochastic matrices. In Proceedings of the 2010 IEEE International Symposium on Information Theory, June 2010.
[2] D.P. Bertsekas, A. Nedić, and A.E. Ozdaglar. Convex analysis and optimization. Athena Scientific, 2003.
[3] J. Chen and A. H. Sayed. Diffusion adaptation strategies for distributed optimization and learning over networks. IEEE Transactions on Signal Processing, 60(8):4289–4305, 2012.
[4] A. D. Dominguez-Garcia and C.N. Hadjicostis. Distributed matrix scaling and application to average consensus on directed graphs. IEEE Transactions on Automatic Control, 58(3):667–681, 2013.
[5] A.D. Dominguez-Garcia and C. Hadjicostis. Distributed strategies for average consensus in directed graphs. In Proceedings of the IEEE Conference on Decision and Control, Dec 2011.
[6] A.D. Dominguez-Garcia, C.N. Hadjicostis, and N.F. Vaidya. Resilient network control of distributed energy resources. IEEE Journal on Selected Areas in Communications, 30(6):1137–1148, 2012.
[7] J.C. Duchi, A. Agarwal, and M.J. Wainwright. Dual averaging for distributed optimization: Convergence analysis and network scaling. IEEE Transactions on Automatic Control, 57(3):592–606, 2012.
[8] B. Gharesifard and J. Cortes. Distributed strategies for making a digraph weight-balanced. In Proceedings of the 47th Annual Allerton Conference on Communication, Control, and Computing, pages 771–777, 2009.
[9] B. Gharesifard and J. Cortes. Distributed continuous-time convex optimization on weight-balanced digraphs. IEEE Transactions on Automatic Control, 59(3):781–786, 2014.
[10] B. Gharesifard, B. Touri, T. Basar, and C. Langbort. Distributed optimization by myopic strategic interactions and the price of heterogeneity. In Proceedings of the 52nd IEEE Conference on Decision and Control, 2013.
[11] B. Johansson. On distributed optimization in networked systems. PhD thesis, Royal Institute of Technology (KTH), rRTA-EE 2008:065, 2008.
[12] B. Johansson, T. Keviczky, M. Johansson, and K.H. Johansson. Subgradient methods and consensus algorithms for solving convex optimization problems. In Proceedings of the IEEE Conference on Decision and Control, 2008.

[13] D. Kempe, A. Dobra, and J. Gehrke. Gossip-based computation of aggregate information. In Proceedings of the 44th Annual IEEE Symposium on Foundations of Computer Science, pages 482–491, Oct. 2003.

[14] H. Li and Z. Han. Competitive spectrum access in cognitive radio networks: graphical game and learning. In IEEE Wireless Communications and Networking Conference, pages 1–6, 2010.

[15] I. Lobel and A. Ozdaglar. Distributed subgradient methods for convex optimization over random networks. IEEE Transactions on Automatic Control, 56(6):1291–1306, June 2011.

[16] I. Lobel, A. Ozdaglar, and D. Feijer. Distributed multi-agent optimization with state-dependent communication. Mathematical Programming, 129(2):255–284, 2011.

[17] C. Lopes and A.H. Sayed. Incremental adaptive strategies over distributed networks. IEEE Transactions on Signal Processing, 55(8):4046–4077, 2007.

[18] J. Lu and C. Y. Tang. Zero-gradient-sum algorithms for distributed convex optimization: The continuous-time case. IEEE Transactions on Automatic Control, 57(9):2348–2354, 2012.

[19] A. Nedić. Asynchronous broadcast-based convex optimization over a network. IEEE Transactions on Automatic Control, 56(6):1337–1351, 2011.

[20] A. Nedić and S. Lee. On stochastic subgradient mirror-descent algorithm with weighted averaging. SIAM Journal on Optimization, 24(1):84–107, 2014.

[21] A. Nedić and A. Olshevsky. Distributed optimization of strongly convex functions over time-varying graphs. In Proceedings of the IEEE Global Conference on Signal and Information Processing, 2013.

[22] A. Nedić and A. Olshevsky. Distributed optimization over time-varying directed graphs. available at: http://arxiv.org/abs/1303.2289, 2013.

[23] A. Nedić and A. Ozdaglar. Distributed subgradient methods for multi-agent optimization. IEEE Transactions on Automatic Control, 54(1):48–61, Jan. 2009.

[24] G. Neglia, G. Reina, and S. Alouf. Distributed gradient optimization for epidemic routing: A preliminary evaluation. In IEEE Wireless Days, 2nd IFIP, pages 1–6, 2009.

[25] A. Olshevsky. Efficient information aggregation for distributed control and signal processing. PhD thesis, MIT, 2010.

[26] M. Rabbat and R.D. Nowak. Distributed optimization in sensor networks. In IPSW, pages 20–27, 2004.

[27] S.S. Ram, V.V. Veeravalli, and A. Nedić. Distributed non-autonomous power control through distributed convex optimization. In IEEE INFOCOM, pages 3001–3005, 2009.

[28] K. Srivastava and A. Nedić. Distributed asynchronous constrained stochastic optimization. IEEE J. Sel. Topics. Signal Process., 5(4):772–790, 2011.

[29] K.I. Tsianos. The role of the Network in Distributed Optimization Algorithms: Convergence Rates, Scalability, Communication / Computation Tradeoffs and Communication Delays. PhD thesis, McGill University, Dept. of Electrical and Computer Engineering, 2013.

[30] K.I. Tsianos, S. Lawlor, and M.G. Rabbat. Consensus-based distributed optimization: Practical issues and applications in large-scale machine learning. In Proceedings of the 50th Allerton Conference on Communication, Control, and Computing, 2012.

[31] K.I. Tsianos, S. Lawlor, and M.G. Rabbat. Push-sum distributed dual averaging for convex optimization. In Proceedings of the IEEE Conference on Decision and Control, 2012.

[32] K.I. Tsianos and M.G. Rabbat. Distributed consensus and optimization under communication delays. In Proc. of Allerton Conference on Communication, Control, and Computing, pages 974–982, 2011.

[33] K.I. Tsianos and M.G. Rabbat. Distributed strongly convex optimization. In Proc. of Allerton Conference on Communication, Control, and Computing, 2012.

[34] K.I. Tsianos and M.G. Rabbat. Simple iteration-optimal distributed optimization. In Proceedings of the European Conference on Signal Processing, 2013.

[35] N. H. Vaidya, C.N. Hadjicostis, and A.D. Domínguez-García. Robust average consensus over packet dropping links: analysis via coefficients of ergodicity. In Proceedings of the 51st Annual Conference on Decision and Control, pages 2761–2766, 2012.

[36] E. Wei and A. Ozdaglar. On the O(1/k) convergence of asynchronous distributed alternating direction method of multipliers. In Proceedings of the IEEE Global Conference on Signal and Information Processing, 2013.

[37] B. Yang and M. Johansson. Distributed optimization and games: a tutorial overview. In Networked Control Systems, 2010.