Distributed Adaptive Convex Optimization on Directed Graphs via Continuous-Time Algorithms

DOI:
10.1109/TAC.2017.2750103

To appear in IEEE Transactions on Automatic Control

Distributed Adaptive Convex Optimization on Directed Graphs via Continuous-Time Algorithms

Zhenhong Li, Zhengtao Ding, *Senior Member, IEEE,* Junyong Sun, and Zhongkui Li, *Member, IEEE*

***Abstract*—This note considers the distributed optimization problem on directed graphs with nonconvex local objective functions and the unknown network connectivity. A new adaptive algorithm is proposed to minimize a differentiable global objective function. By introducing dynamic coupling gains and updating the coupling gains using relative information of system states, the nonconvexity of local objective functions, unknown network connectivity and the uncertain dynamics caused by locally Lipschitz gradients are tackled concurrently. Consequently, the global asymptotic convergence is established when the global objective function is strongly convex and the gradients of local objective functions are only locally Lipschitz. When the communication graph is strongly connected and weight-balanced, the algorithm is independent of any global information. Then, the algorithm is naturally extended to unbalanced directed graphs by using the left eigenvector of the Laplacian matrix associated with the zero eigenvalue. Several numerical simulations are presented to verify the results.**

***Index Terms*—Consensus control, distributed convex optimization, adaptive control.**

I. Introduction

The last decades have witnessed a growing interest of research in distributed optimization, due to its potential applications in a variety of scenarios such as sensor networks, distributed parameters estimation, power system economic dispatch and regression of distributed data (see, e.g., [1]–[4]). An important class of distributed optimization problems is to minimize a global objective function which is the sum of local objective functions, by local computation and information exchange with neighboring agents. This kind of distributed optimization problems have been addressed by many researchers from various perspectives (see, e.g., [5]–[20]).

Most of existing algorithms are based on discrete-time dynamics (see e.g., [5]–[10], [21]). By designing the consensus-based dynamics, these discrete-time algorithms can find the solution of the optimization problem. Recently, continuous-time algorithms have been introduced to solve distributed optimization problems (see, e.g., [11]–[20]). In [14], [16] and [19], the Newton-Raphson and the Zero-Gradient-Sum based continuous-time algorithms achieve the global convergence on undirected graphs using the positive bounded Hessian of local objective functions. Since the requirement of the positive bounded Hessian, local objective functions are assumed to be twice differentiable and convex. The projection based algorithm in [15] removes the requirement of the twice differentiability of local objective functions by using the projection of their gradients. Two adaptive schemes are designed in [20] to solve the distributed optimization problem for general linear dynamics with undirected communications. The global convergence is established when local objective functions are convex and local gradients are error-bounded. The algorithms in [12], [13] and [17] successfully solve the distributed optimization problem on weight-balanced directed graphs, which is more challenging than the undirected case. To deal with the unidirectional gradient flow, the global Lipschitz constants of local gradients and the network connectivity (i.e., the smallest nonzero eigenvalue of the Laplacian matrix) are used in the algorithms. However, these parameters require the knowledge of entire network connections and are difficult to get for large scale networks. The global asymptomatic convergence of the algorithms is established when all local objective functions with global Lipschitz gradients are convex. Moreover, if the gradients of local objective functions are only locally Lipschitz, the global convergence will degrade to semiglobal.

Two main challenges for the distributed optimization with directed communications are 1) removing the requirement of global information, and establishing global convergence with only locally Lipschitz gradients; 2) relaxing the common assumption (see, e.g., [11]–[20], [22]) of convexity of local objective functions. It is well-known in the consensus control design that adaptive techniques can be used to deal with the unknown network connectivity (see, e.g., [23]–[25]). However, different from simply borrowing the adaptive techniques from consensus control design, the design of adaptive schemes in the distributed optimization has to tackle different features: the non-linearity of local gradients and the coupled dynamics between system states and internal states caused by asymmetric communications.

In this note, we propose a new adaptive algorithm on weight-balanced directed graphs. Consequently, the requirements of the Lipschitz constants and the network connectivity are removed. Unlike the previous results, the convexity properties of local objective functions is not used in our convergence analysis, which makes local objective functions allowed to be nonconvex. The global asymptotic convergence can be guaranteed if the sum of local objective functions is strongly convex. Another contribution of this note is that we extend our results to general unbalanced directed graphs. The global asymptotic convergence can be guaranteed on

Z. Li and Z. Ding are with School of Electrical and Electronic Engineering, University of Manchester, Sackville Street Building, Manchester M13 9PL, UK (emails: zhenhong.li@postgrad.manchester.ac.uk; zhengtao.ding@manchester.ac.uk).

J. Sun and Z. Li are with State Key Laboratory for Turbulence and Complex Systems, Department of Mechanics and Engineering Science, College of Engineering, Peking University, Beijing 100871, China (e-mails: sjymath@pku.edu.cn; zhongkli@pku.edu.cn).

unbalanced directed graphs provided that the left eigenvector of the Laplacian matrix associated with the zero eigenvalue is available.

The remaining sections of this note are organized as follows. Section II is devoted to notations and mathematical preliminaries. In Section III, the distributed optimization problem is formulated. In Section IV, an adaptive distributed algorithm is proposed for strongly connected weight-balanced directed graphs. In Section V, a different adaptive algorithm is designed for unbalanced directed graphs. Simulations are included in Section VI, and conclusions are drawn in the last section.

II. Notations and Preliminaries

Throughout this section, we introduce our notations and some basic concepts of convex functions and graph theory.

A. Notations

Let $\mathbb{R}$, $\mathbb{R}^n$, $\mathbb{R}^{n\times m}$ denote the sets of real numbers, real vectors of dimension n, and real matrices of size $n\times m$, respectively. $\mathbb{R}_{>0}$ denotes the positive real numbers. The superscript T denotes the transpose of a real matrix. The identity matrix of dimension n is denoted by I_n, and the column vector of size n with all entries equal to one is denoted by $\mathbf{1}_n$. For a vector $a\in\mathbb{R}^n$, $\|a\|$ is the Euclidean norm of a; for a matrix $A\in\mathbb{R}^{n\times n}$, $|||A|||$ is the spectral norm of A (also known as its maximum singular value). The ith eigenvalue of the matrix A, is denoted by $\lambda_i(A)$. Besides, the symbol $\otimes$ denotes the Kronecker product of the matrices, which has the properties that $(B\otimes C)(D\otimes E)=(BD)\otimes(CE)$, $(B\otimes C)^{\mathrm{T}}=B^{\mathrm{T}}\otimes C^{\mathrm{T}}$, where B, C, D, E are matrices with proper dimensions. For a differentiable function $f:\mathbb{R}^n\to\mathbb{R}$, ∇f denotes the gradient of f; f is strongly convex over a convex set $\Omega\subseteq\mathbb{R}^n$ iff there exists $m\in\mathbb{R}_{>0}$ such that $(x-y)^{\mathrm{T}}(\nabla f(x)-\nabla f(y))>m\|x-y\|^2$ for all $x,y\in\Omega$, $x\neq y$; f is locally Lipschitz at $x\in\mathbb{R}^n$ if there exists a neighborhood $\mathcal{W}$ of x and $M\in\mathbb{R}_{\geq 0}$ such that $|f(y)-f(z)|\leq M\|y-z\|$, for $y,z\in\mathcal{W}$; f is locally Lipschitz on $\mathbb{R}^n$ if it is locally Lipschitz at x for all $x\in\mathbb{R}^n$.

B. Graph Theory

The information flow among agents is described by a directed graph. Let a triplet $\mathcal{G}=(\mathcal{V},\mathcal{E},\mathcal{A})$ be a directed graph, where $\mathcal{V}=\{1,\ldots,N\}$ is a set of nodes, $\mathcal{E}\subseteq\mathcal{V}\times\mathcal{V}$ is a set of edges, and $\mathcal{A}=[a_{ij}]\in\mathbb{R}^{N\times N}$ is a weighted adjacency matrix. An edge $(i,j)\in\mathcal{E}$ represents that ith agent can receive the information from jth agent, but not vice versa. The jth agent is a neighbor of ith agent if $(i,j)\in\mathcal{E}$. A directed path from node i_1 to node i_q is a sequence of ordered edges in the form of $(i_1,i_2),\ldots,(i_{q-1},i_q)$. A directed graph is strongly connected if there exists a directed path connecting every pair of nodes. The weighted adjacency matrix $\mathcal{A}$ is defined as $a_{ij}>0$ if $(i,j)\in\mathcal{E}$, otherwise $a_{ij}=0$. Due to the fact that there is no self-loop in graph, $a_{ii}=0$ for all nodes. The Laplacian matrix $\mathcal{L}=[l_{ij}]\in\mathbb{R}^{N\times N}$ associated with the directed graph $\mathcal{G}$ is defined as $l_{ii}=\sum_{j=1}^N a_{ij}$ and $l_{ij}=-a_{ij}$ for $i\neq j$. The eigenvalues of a symmetric $\mathcal{L}$ can be ordered as $0\leq\lambda_2(\mathcal{L})\leq\cdots\leq\lambda_N(\mathcal{L})$. A directed graph $\mathcal{G}$ is weight-balanced iff $\mathbf{1}_N^{\mathrm{T}}\mathcal{L}=\mathbf{0}_N^{\mathrm{T}}$.

Lemma 1: ([26]) Let $\mathcal{L}\in\mathbb{R}^{N\times N}$ be the Laplacian matrix of a strongly connected directed graph $\mathcal{G}$. The following properties hold:

1) Matrix $\mathcal{L}$ has a simple zero eigenvalue corresponding to the right eigenvector $\mathbf{1}_N$, and all nonzero eigenvalues have positive real part.
2) Let $r=[r_1,r_2,\ldots,r_N]^{\mathrm{T}}$, $r_i\in\mathbb{R}_{>0}$, $i=1,2,\ldots,N$, be the left eigenvector of $\mathcal{L}$ associated with the zero eigenvalue and $R=\mathrm{diag}(r_1,r_2,\ldots,r_N)$. Then, $\min_{\zeta^{\mathrm{T}}x=0,x\neq\mathbf{0}}\frac{x^{\mathrm{T}}\bar{\mathcal{L}}x}{x^{\mathrm{T}}x}>\frac{\lambda_2(\bar{\mathcal{L}})}{N}$, where $\bar{\mathcal{L}}\triangleq R\mathcal{L}+\mathcal{L}^{\mathrm{T}}R$, ζ is any vector with positive entries. Moreover, $r=\mathbf{1}_N$ iff $\mathcal{G}$ is strongly connected and weight-balanced.

III. Problem Statement and Equivalent Formulations

In this section, we consider a set of N agents interacting over a directed connection graph. Each agent has a local cost function $f_i:\mathbb{R}^n\to\mathbb{R}$. The global cost function is defined as $f(z)=\sum_{i=1}^N f_i(z)$. The objective of this note is to design a continuous-time distributed algorithm such that each agent can solve the optimization problem

$$\min_{z\in\mathbb{R}^n} f(z), \tag{1}$$

by using its own and neighboring information.

Following assumptions are supposed to be satisfied throughout this note.

Assumption 1: The global cost function f is differentiable and strongly convex over $\mathbb{R}^n$. The local cost function f_i is differentiable and its gradient is locally Lipschitz on $\mathbb{R}^n$, i.e., for any compact set $\mathcal{U}\subset\mathbb{R}^n$, there always exists $M_i\in\mathbb{R}_{\geq 0}$ such that $|\nabla f_i(y)-\nabla f_i(z)|\leq M_i\|y-z\|$ for $y,z\in\mathcal{U}$.

From Assumption 1, the strong convexity of the global cost function f guarantees the unique solution of the problem (1).

Assumption 2: The communication graph $\mathcal{G}$ is strongly connected.

Under Assumption 2, the problem (1) can be reformulated as

$$\min_{x_i\in\mathbb{R}^n}\tilde{f}=\sum_{i=1}^N f_i(x_i),\ \text{subject to } (\mathcal{L}\otimes I_n)x=\mathbf{0}, \tag{2}$$

where x_i is the state of ith agent, and $x=[x_1^{\mathrm{T}},x_2^{\mathrm{T}},\ldots,x_N^{\mathrm{T}}]^{\mathrm{T}}$ is the state of network. In view of Lemma 1, $(\mathcal{L}\otimes I_n)x=\mathbf{0}$ iff $x=\mathbf{1}_N\otimes\tau$, for some $\tau\in\mathbb{R}^n$. Then, it can be concluded that the problem (1) is equivalent to the problem (2). By reformulating the problem (1), the problem is transformed into a distributed minimization problem under a consensus condition.

IV. Distributed Adaptive Continuous-Time Convex Optimization Algorithm On Weight-Balanced Directed Graphs

In this section, we propose a fully distributed algorithm to solve the problem (2). Consider the following algorithm with dynamic coupling gains

$$\dot{v}_i = \gamma_1(\alpha_i+\beta_i)\sum_{j=1}^{N} a_{ij}(x_i-x_j), \tag{3a}$$

$$\dot{x}_i = -\gamma_2\nabla f_i(x_i) - \gamma_1(\alpha_i+\beta_i)\sum_{j=1}^{N} a_{ij}(x_i-x_j) - \sum_{j=1}^{N} a_{ij}(v_i-v_j), \tag{3b}$$

$$\dot{\alpha}_i = e_i^{\mathrm{T}} e_i, \tag{3c}$$

where $\beta_i = e_i^{\mathrm{T}} e_i$, $v_i \in \mathbb{R}^n$ is the internal state of the algorithm, $e_i = \sum_{j=1}^{N} a_{ij}(x_i-x_j)$ is the relative error, $\gamma_1, \gamma_2 \in \mathbb{R}_{>0}$ are control gains, and α_i, β_i are the dynamic coupling gains with $\alpha_i(0) \in \mathbb{R}_{>0}$.

Note that (3) is distributed, since each agent only communicates with its neighboring agents. The term $\sum_{j=1}^{N} a_{ij}(v_i-v_j)$ in (3b) implies that agents need to transmit the internal states of the algorithm via the communication graph $\mathcal{G}$.

The dynamics of the network can be written in a compact form as

$$\dot{v} = \gamma_1[(\hat{\alpha}+\hat{\beta})\mathcal{L}\otimes I_n]x, \tag{4a}$$

$$\dot{x} = -\gamma_2\nabla\tilde{f}(x) - \gamma_1[(\hat{\alpha}+\hat{\beta})\mathcal{L}\otimes I_n]x - (\mathcal{L}\otimes I_n)v, \tag{4b}$$

$$\dot{\alpha}_i = e_i^{\mathrm{T}} e_i, \tag{4c}$$

where $\beta_i = e_i^{\mathrm{T}} e_i$, $\hat{\alpha} = \mathrm{diag}(\alpha_1,\alpha_2,\ldots,\alpha_N)$, $\hat{\beta} = \mathrm{diag}(\beta_1,\beta_2,\ldots,\beta_N)$, $v = [v_1^{\mathrm{T}}, v_2^{\mathrm{T}}, \ldots, v_N^{\mathrm{T}}]^{\mathrm{T}}$ is the internal state of the network, and $\nabla\tilde{f}(x) = [\nabla f_1(x_1)^{\mathrm{T}}, \nabla f_2(x_2)^{\mathrm{T}}, \cdots, f_N(x_N)^{\mathrm{T}}]^{\mathrm{T}}$ is the vector of the network gradient.

Lemma 2: Under Assumptions 1 and 2 , if the communication graph $\mathcal{G}$ is weight-balanced, the equilibrium point of (4) is an optimal solution of the distributed optimization problem (2).

Proof: We can obtain the equilibrium point $(\tilde{x},\tilde{v})$ of (4), from

$$\mathbf{0} = \gamma_1[(\hat{\alpha}+\hat{\beta})\mathcal{L}\otimes I_n]\tilde{x}, \tag{5}$$

$$\mathbf{0} = -\gamma_2\nabla\tilde{f}(\tilde{x}) - \gamma_1[(\hat{\alpha}+\hat{\beta})\mathcal{L}\otimes I_n]\tilde{x} - (\mathcal{L}\otimes I_n)\tilde{v}. \tag{6}$$

In the sequel, we will show that the equilibrium point is a solution of the problem (2). Deducing from (5) and (6), the equilibrium point satisfies

$$\tilde{x}_i = x^\star, i = 1, 2, \ldots, N, \tag{7}$$

$$\sum_{j=1}^{N} a_{ij}(\tilde{v}_i - \tilde{v}_j) = -\gamma_2\nabla f_i(x^\star), \tag{8}$$

$$\gamma_2\sum_{i=1}^{N}\nabla f_i(x^\star) = (\mathbf{1}_N^{\mathrm{T}}\mathcal{L}\otimes I_n)\tilde{v} = \mathbf{0}, \tag{9}$$

where $x^\star \in \mathbb{R}^n$. Since $\tilde{f}$ is strongly convex, $\sum_{i=1}^{N}\nabla f_i(x^\star) = \mathbf{0}$ implies that $(\tilde{x},\tilde{v})$ is a global minimizer of (2). Note that if $(\tilde{x},\tilde{v})$ is a solution of (2), so is $(\tilde{x},\tilde{v}+\mathbf{1}_N\otimes\kappa), \kappa\in\mathbb{R}^n$. ■

Theorem 1: Under Assumptions 1 and 2 , if the communication graph $\mathcal{G}$ is weight-balanced, the dynamic algorithm (3) solves the distributed optimization problem (2) for any $x_i(0), v_i(0) \in \mathbb{R}^n$. Furthermore, the dynamic coupling gains α_i converge to some finite steady-state values.

Proof: Theorem 1 is proved by showing that the trajectories of (x,v) converge to the equilibrium point of (4). Transferring the equilibrium point $(\tilde{x},\tilde{v})$ to the origin by the state transformation $\mu = v-\tilde{v}, g = x-\tilde{x}$, we can further write the network dynamics as

$$\dot{\mu} = \gamma_1[(\hat{\alpha}+\hat{\beta})\mathcal{L}\otimes I_n]g, \tag{10a}$$

$$\dot{g} = -\gamma_2(\nabla\tilde{f}(g+\tilde{x}) - \nabla\tilde{f}(\tilde{x})) - \gamma_1[(\hat{\alpha}+\hat{\beta})\mathcal{L}\otimes I_n]g - (\mathcal{L}\otimes I_n)\mu, \tag{10b}$$

$$\dot{\alpha}_i = \left[\sum_{j=1}^{N} a_{ij}(g_i-g_j)\right]^{\mathrm{T}}\left[\sum_{j=1}^{N} a_{ij}(g_i-g_j)\right], \tag{10c}$$

where $\beta_i = \left[\sum_{j=1}^{N} a_{ij}(g_i-g_j)\right]^{\mathrm{T}}\left[\sum_{j=1}^{N} a_{ij}(g_i-g_j)\right]$. To get (10), we have used $(\mathcal{L}\otimes I_n)[(\hat{\alpha}+\hat{\beta})\otimes I_n](\mathcal{L}\otimes I_n) = [\mathcal{L}(\hat{\alpha}+\hat{\beta})\otimes I_n](\mathcal{L}\otimes I_n)$.

The distributed optimization problem (2) is solved by the algorithm (3) if $\lim_{t\to\infty}\mu(t) = \mathbf{1}_N\otimes\kappa, \kappa\in\mathbb{R}^n$ and $\lim_{t\to\infty} g(t) = \mathbf{0}$.

Introducing another state transformation $\varrho = (\mathcal{L}\otimes I_n)\mu$, $\eta = (\mathcal{L}\otimes I_n)g$, we obtain the dynamics

$$\dot{\varrho} = \gamma_1[\mathcal{L}(\hat{\alpha}+\hat{\beta})\otimes I_n]\eta, \tag{11a}$$

$$\dot{\eta} = -\gamma_2(\mathcal{L}\otimes I_n)h - \gamma_1[\mathcal{L}(\hat{\alpha}+\hat{\beta})\otimes I_n]\eta - (\mathcal{L}\otimes I_n)\varrho, \tag{11b}$$

$$\dot{\alpha}_i = \eta_i^{\mathrm{T}}\eta_i, \tag{11c}$$

where $h = \nabla\tilde{f}(g+\tilde{x}) - \nabla\tilde{f}(\tilde{x})$ and $\beta_i = \eta_i^{\mathrm{T}}\eta_i$.

Consider following positive definite functions

$$V_1 = \frac{1}{2}\sum_{i=1}^{N}(\alpha_i-\alpha)^2, \tag{12}$$

$$V_2 = \frac{1}{2}\sum_{i=1}^{N}(2\alpha_i+\beta_i)\eta_i^{\mathrm{T}}\eta_i, \tag{13}$$

where α is a positive scalar to be designed later. The time derivatives of (12) and (13) along the trajectory of (11) are given by

$$\dot{V}_1 = \sum_{i=1}^{N}(\alpha_i-\alpha)e_i^{\mathrm{T}}e_i = \sum_{i=1}^{N}\eta_i^{\mathrm{T}}(\alpha_i-\alpha)\eta_i = \eta^{\mathrm{T}}[(\hat{\alpha}-\alpha I_N)\otimes I_n]\eta, \tag{14}$$

$$\begin{aligned}\dot{V}_2 &= \sum_{i=1}^{N}(2\alpha_i+2\beta_i)\eta_i^{\mathrm{T}}\dot{\eta}_i + \beta_i\dot{\alpha}_i \\ &= -2\gamma_2\eta^{\mathrm{T}}[(\hat{\alpha}+\hat{\beta})\mathcal{L}\otimes I_n]h \\ &\quad - 2\eta^{\mathrm{T}}[(\hat{\alpha}+\hat{\beta})\mathcal{L}\otimes I_n]\varrho + \eta^{\mathrm{T}}(\hat{\beta}\otimes I_n)\eta \\ &\quad - \gamma_1\eta^{\mathrm{T}}\{[(\hat{\alpha}+\hat{\beta})(\mathcal{L}+\mathcal{L}^{\mathrm{T}})(\hat{\alpha}+\hat{\beta})]\otimes I_n\}\eta.\end{aligned} \tag{15}$$

Let $\tilde{\eta} = [(\hat{\alpha}+\hat{\beta})\otimes I_n]\eta$. Then, we can obtain $\tilde{\eta}^{\mathrm{T}}[(\hat{\alpha}+\hat{\beta})^{-1}\mathbf{1}_N\otimes\mathbf{1}_n] = g^{\mathrm{T}}(\mathcal{L}^{\mathrm{T}}\mathbf{1}_N\otimes\mathbf{1}_n) = 0$.

Since all the entries of $(\hat{\alpha}+\hat{\beta})^{-1}\mathbf{1}_N\otimes\mathbf{1}_n$ are positive, in light of Lemma 1, it follows that

$$\eta^{\mathrm{T}}\{[(\hat{\alpha}+\hat{\beta})\mathcal{L}(\hat{\alpha}+\hat{\beta})+(\hat{\alpha}+\hat{\beta})\mathcal{L}^{\mathrm{T}}(\hat{\alpha}+\hat{\beta})]\otimes I_n\}\eta \geq \frac{\lambda_2(\hat{\mathcal{L}})}{N}\eta^{\mathrm{T}}[(\hat{\alpha}+\hat{\beta})^2\otimes I_n]\eta,$$

where $\hat{\mathcal{L}}=\mathcal{L}+\mathcal{L}^{\mathrm{T}}$ and the equality holds iff $\eta=\mathbf{0}$. By incorporating this fact into (15), we have

$$\begin{aligned}\dot{V}_1+\dot{V}_2\leq&-\frac{\gamma_1\lambda_2(\hat{\mathcal{L}})}{N}\eta^{\mathrm{T}}[(\hat{\alpha}+\hat{\beta})^2\otimes I_n]\eta\\&+\eta^{\mathrm{T}}[(\hat{\alpha}+\hat{\beta}-\alpha I_N)\otimes I_n]\eta-2\eta^{\mathrm{T}}[(\hat{\alpha}+\hat{\beta})\mathcal{L}\\&\otimes I_n]\varrho-2\gamma_2\eta^{\mathrm{T}}[(\hat{\alpha}+\hat{\beta})\mathcal{L}\otimes I_n]h.\end{aligned}\tag{16}$$

Define a convex set containing $(\tilde{x},\tilde{v})$ as

$$\mathcal{H}=\{(x,v)\in\mathbb{R}^{Nn}\times\mathbb{R}^{Nn}\mid \|x-\tilde{x}\|\leq\|x(0)-\tilde{x}\|\,,\ \text{for any } v\in\mathbb{R}^{Nn}\}.$$

Since $\tilde{x}$ is unique, $\mathcal{H}$ is compact for x. Based on Assumption 1, there exists $M_i\in\mathbb{R}_{>0}$ such that $\|\nabla f_i(x_i)-\nabla f_i(\tilde{x}_i)\|\leq M_i\|x_i-\tilde{x}_i\|, i=1,2,\ldots,N$, for $(x,v)\in\mathcal{H}$. In what follows, we show that η and ϱ converge to 0. Using Young's inequality, we deduce that

$$-2\eta^{\mathrm{T}}[(\hat{\alpha}+\hat{\beta})\mathcal{L}\otimes I_n]\varrho\leq\frac{\gamma_1\lambda_2(\hat{\mathcal{L}})}{4N}\eta^{\mathrm{T}}[(\hat{\alpha}+\hat{\beta})^2\otimes I_n]\eta+\frac{4N\lambda_N(\mathcal{L}^{\mathrm{T}}\mathcal{L})}{\gamma_1\lambda_2(\hat{\mathcal{L}})}\varrho^{\mathrm{T}}\varrho,\tag{17}$$

$$-2\gamma_2\eta^{\mathrm{T}}[(\hat{\alpha}+\hat{\beta})\mathcal{L}\otimes I_n]h\leq\frac{\gamma_1\lambda_2(\hat{\mathcal{L}})}{2N}\eta^{\mathrm{T}}[(\hat{\alpha}+\hat{\beta})^2\otimes I_n]\eta+\frac{2N\gamma_2^2M_{\max}^2\lambda_N(\mathcal{L}^{\mathrm{T}}\mathcal{L})}{\gamma_1\lambda_2(\hat{\mathcal{L}})}g^{\mathrm{T}}g,\tag{18}$$

where $M_{\max}=\max(M_1,M_2,\ldots,M_N)$. To obtain (18), we have applied the fact

$$\begin{aligned}h^{\mathrm{T}}(\mathcal{L}^{\mathrm{T}}\mathcal{L}\otimes I_n)h&\leq\left|\left|\left|\mathcal{L}^{\mathrm{T}}\mathcal{L}\right|\right|\right|\|h\|^2\\&\leq M_{\max}^2\lambda_N(\mathcal{L}^{\mathrm{T}}\mathcal{L})g^{\mathrm{T}}g.\end{aligned}\tag{19}$$

Based on (16), (17) and (18), we can obtain

$$\begin{aligned}\dot{V}_1+\dot{V}_2\leq&-\frac{\gamma_1\lambda_2(\hat{\mathcal{L}})}{4N}\eta^{\mathrm{T}}[(\hat{\alpha}+\hat{\beta})^2\otimes I_n]\eta\\&+\eta^{\mathrm{T}}[(\hat{\alpha}+\hat{\beta}-\alpha I_N)\otimes I_n]\eta+\frac{4N\lambda_N(\mathcal{L}^{\mathrm{T}}\mathcal{L})}{\gamma_1\lambda_2(\hat{\mathcal{L}})}\varrho^{\mathrm{T}}\varrho\\&+\frac{2N\gamma_2^2M_{\max}^2\lambda_N(\mathcal{L}^{\mathrm{T}}\mathcal{L})}{\gamma_1\lambda_2(\hat{\mathcal{L}})}g^{\mathrm{T}}g.\end{aligned}\tag{20}$$

Consider

$$V_3=\frac{1}{2}(\varrho+\eta)^{\mathrm{T}}(\varrho+\eta).\tag{21}$$

The time derivative of V_3 along (11) can be written as

$$\begin{aligned}\dot{V}_3=&\varrho^{\mathrm{T}}\dot{\varrho}+\eta^{\mathrm{T}}\dot{\eta}+\varrho^{\mathrm{T}}\dot{\eta}+\eta^{\mathrm{T}}\dot{\varrho}\\=&-\gamma_2\eta^{\mathrm{T}}(\mathcal{L}\otimes I_n)h-\eta^{\mathrm{T}}(\mathcal{L}\otimes I_n)\varrho-\gamma_2\varrho^{\mathrm{T}}(\mathcal{L}\otimes I_n)h\\&-\varrho^{\mathrm{T}}(\mathcal{L}\otimes I_n)\varrho\\\leq&\frac{\lambda_2(\hat{\mathcal{L}})+2\lambda_N(\mathcal{L}^{\mathrm{T}}\mathcal{L})}{\lambda_2(\hat{\mathcal{L}})}\eta^{\mathrm{T}}\eta-\frac{\lambda_2(\hat{\mathcal{L}})}{4}\varrho^{\mathrm{T}}\varrho\\&+\frac{\gamma_2^2M_{\max}^2\lambda_N(\mathcal{L}^{\mathrm{T}}\mathcal{L})(\lambda_2(\hat{\mathcal{L}})+8)}{4\lambda_2(\hat{\mathcal{L}})}g^{\mathrm{T}}g.\end{aligned}\tag{22}$$

To get (22), we have used (19) and the facts that

$$\varrho^{\mathrm{T}}(\mathcal{L}\otimes I_n)\varrho=\frac{\varrho^{\mathrm{T}}(\hat{\mathcal{L}}\otimes I_n)\varrho}{2},\quad\lambda_i(\mathcal{L}\mathcal{L}^{\mathrm{T}})=\lambda_i(\mathcal{L}^{\mathrm{T}}\mathcal{L}).$$

Consider a Lyapunov function candidate for the whole closed-loop system as

$$V=V_1+V_2+\frac{17N\lambda_N(\mathcal{L}^{\mathrm{T}}\mathcal{L})}{\gamma_1\lambda_2(\hat{\mathcal{L}})^2}V_3.\tag{23}$$

Applying the results (20) and (22), and from (23), we have

$$\begin{aligned}\dot{V}\leq&\eta^{\mathrm{T}}\left[\left(-\frac{\gamma_1\lambda_2(\hat{\mathcal{L}})}{4N}(\hat{\alpha}+\hat{\beta})^2+(\hat{\alpha}+\hat{\beta})-\alpha I_N\right.\right.\\&\left.\left.+\frac{17N\lambda_N(\mathcal{L}^{\mathrm{T}}\mathcal{L})(\lambda_2(\hat{\mathcal{L}})+2\lambda_N(\mathcal{L}^{\mathrm{T}}\mathcal{L}))}{\gamma_1\lambda_2(\hat{\mathcal{L}})^3}I_N\right)\otimes I_n\right]\eta\\&-\frac{N\lambda_N(\mathcal{L}^{\mathrm{T}}\mathcal{L})}{4\gamma_1\lambda_2(\hat{\mathcal{L}})}\varrho^{\mathrm{T}}\varrho+\left(\frac{2N\gamma_2^2M_{\max}^2\lambda_N(\mathcal{L}^{\mathrm{T}}\mathcal{L})}{\gamma_1\lambda_2(\hat{\mathcal{L}})}\right.\\&\left.+\frac{17N\gamma_2^2M_{\max}^2\lambda_N(\mathcal{L}^{\mathrm{T}}\mathcal{L})^2(\lambda_2(\hat{\mathcal{L}})+8)}{4\gamma_1\lambda_2(\hat{\mathcal{L}})^3}\right)g^{\mathrm{T}}g.\end{aligned}\tag{24}$$

Let $\delta\in\mathbb{R}_{>0}$ be an arbitrary small positive constant. Since $g(x(t))^{\mathrm{T}}g(x(t))\leq g(x(0))^{\mathrm{T}}g(x(0))$, for $(x,v)\in\mathcal{H}$, when $\eta^{\mathrm{T}}\eta\geq\delta$, there always exists a sufficiently large positive scalar $\epsilon\in\mathbb{R}_{>0}$ such that $\epsilon\eta^{\mathrm{T}}\eta\geq\left(\frac{2N\gamma_2^2M_{\max}^2\lambda_N(\mathcal{L}^{\mathrm{T}}\mathcal{L})}{\gamma_1\lambda_2(\hat{\mathcal{L}})}+\frac{17N\gamma_2^2M_{\max}^2\lambda_N(\mathcal{L}^{\mathrm{T}}\mathcal{L})^2(\lambda_2(\hat{\mathcal{L}})+8)}{4\gamma_1\lambda_2(\hat{\mathcal{L}})^3}\right)g^{\mathrm{T}}g$.

Choosing $\alpha\geq\frac{17N\lambda_N(\mathcal{L}^{\mathrm{T}}\mathcal{L})(\lambda_2(\hat{\mathcal{L}})+2\lambda_N(\mathcal{L}^{\mathrm{T}}\mathcal{L}))}{\gamma_1\lambda_2(\hat{\mathcal{L}})^3}+\epsilon+1+\frac{N}{\gamma_1\lambda_2(\hat{\mathcal{L}})}$,we can obtain from (24)

$$\begin{aligned}\dot{V}\leq&\eta^{\mathrm{T}}\left[-\frac{\gamma_1\lambda_2(\hat{\mathcal{L}})}{4N}\left((\hat{\alpha}+\hat{\beta})-\frac{2N}{\gamma_1\lambda_2(\hat{\mathcal{L}})}I_N\right)^2\otimes I_n\right]\eta\\&-\eta^{\mathrm{T}}\eta-\frac{N\lambda_N(\mathcal{L}^{\mathrm{T}}\mathcal{L})}{4\gamma_1\lambda_2(\hat{\mathcal{L}})}\varrho^{\mathrm{T}}\varrho.\end{aligned}\tag{25}$$

For the reason that $\dot{V}$ is continuous and negative definite for $\eta^{\mathrm{T}}\eta\geq\delta$, $V(t)$ is bounded and α_i converge to some positive values. Applying LaSalle's invariance principle, we can conclude that ϱ converges asymptotically to zero, and η converges to a residual set $\mathcal{D}=\left\{\eta|\,\|\eta\|^2\leq\delta\right\}$. Since δ can be chosen as an arbitrary small constant, we assume η converges to zero. It then follows that (x,v) converges to the set $\mathcal{I}=\{(x,v)\in\mathcal{H}\mid x=\tilde{x}+\mathbf{1}_N\otimes\xi, v=\tilde{v}+\mathbf{1}_N\otimes\kappa,\xi,\kappa\in\mathbb{R}^n\}$ as $t\to\infty$. Note that ϵ, δ and α are auxiliary variables only used for convergence analysis. These variables are not the parameters in the algorithm (3).

In what follows, we prove $\xi = \mathbf{0}_n$ by seeking a contradiction. Assume $\xi \neq \mathbf{0}_n$, based on (9) and (3b), the dynamics of ξ can be written as

$$\begin{aligned}\dot{\xi} &= \frac{1}{N}(\mathbf{1}_N^{\mathrm{T}} \otimes I_n)(\dot{\tilde{x}} + \mathbf{1}_N \otimes \dot{\xi}) = \frac{1}{N}(\mathbf{1}_N^{\mathrm{T}} \otimes I_n)\dot{x} \\ &= -\frac{\gamma_2}{N}\sum_{i=1}^{N} \nabla f_i(x^\star + \xi) + \frac{\gamma_2}{N}\sum_{i=1}^{N} \nabla f_i(x^\star) \\ &= -\frac{\gamma_2}{N}\nabla f(x^\star + \xi). \end{aligned} \tag{26}$$

From (26), we can deduce that ξ moves towards to the point which satisfies $\nabla f(x^\star + \xi) = \mathbf{0}$. For the reason that f is strongly convex, the critical point $x^\star$ of f is unique, which, since $\xi \neq \mathbf{0}$, is a contradiction.

Finally, we can conclude that the trajectories of (3) which start from $x(0), v(0) \in \mathbb{R}^{Nn}$ converge to the global minimizer $(\tilde{x}, \tilde{v} + \mathbf{1}_N \otimes \kappa)$, for some $\kappa \in \mathbb{R}^n$. The algorithm (3) solves the distributed convex optimization problem. ■

Remark 1: For any $x_i(0), v_i(0) \in \mathbb{R}^n$, we can always define a convex set $\mathcal{H}$ such that all the trajectories of algorithm (3) converge to the global minimizer set $\mathcal{I} = \{(x, v) \in \mathcal{H} \mid x = \tilde{x}, v = \tilde{v} + \mathbf{1}_N \otimes \kappa, \kappa \in \mathbb{R}^n\}$. The convergence of (3) is global, while the results shown in [12], [13] can only achieve semiglobal convergence when $\nabla f_i, i = 1, 2, \cdots, N$, are locally Lipschitz.

Remark 2: The global convergence of the algorithm can be guaranteed for any $\gamma_1, \gamma_2 \in \mathbb{R}_{>0}$. The values of γ_1, γ_2 and $\alpha_i(0)$ can be used to improve the transient process of the algorithm. The static control gains γ_1, γ_2 can be interpreted as the weights of local gradient and agent connectivity, respectively; increasing γ_1 and γ_2 will increase convergence rate, however, high static control gains may cause more oscillations. The transient performances of the proposed algorithms also are related to the static control gain ratio $\frac{\min(\alpha_i(0))\gamma_1}{\gamma_2}$. By increasing the ratio $\frac{\min(\alpha_i(0))\gamma_1}{\gamma_2}$, the consensus effects will be enhanced.

Remark 3: Many previous results have been obtained for the undirected connected graph, but the value of network connectivity is required when the communication graph is a weight-balanced directed graph (e.g., [12], [13]). By introducing the dynamic coupling gains α_i and β_i and continuously updating the coupling gains using the relative errors e_i, we solve the distributed optimization without using the network connectivity. Of course, algorithm (3) can also be applied to undirected connected graphs.

Remark 4: In Assumption 1, only the global cost function is required to be strictly convex, while the local cost functions f_i can be any differentiable functions. Note that the local cost functions are also assumed to be strongly convex in [12]–[15], [22]. Different from [14], there is no restriction on the local cost functions f_i to be twice differentiable with the proposed scheme.

V. Distributed Adaptive Continuous-Time Convex Optimization Algorithm On Unbalanced Directed Graphs

In the previous section, a distributed adaptive algorithm is proposed to solve the problem (2) with strongly connected and weight-balanced graphs. Here, we extend our analysis for strongly connected unbalanced graphs. The distributed optimization algorithm with dynamic coupling gains can be designed as

$$\dot{v}_i = \gamma_1(\alpha_i + \beta_i)\sum_{j=1}^{N} a_{ij}(x_i - x_j), \tag{27a}$$

$$\dot{x}_i = -\gamma_2 \nabla \bar{f}_i(x_i) - \gamma_1(\alpha_i + \beta_i)\sum_{j=1}^{N} a_{ij}(x_i - x_j) - \sum_{j=1}^{N} a_{ij}(v_i - v_j), \tag{27b}$$

$$\dot{\alpha}_i = e_i^{\mathrm{T}} e_i, \tag{27c}$$

where $\beta_i = e_i^{\mathrm{T}} e_i$, $\nabla \bar{f}_i(x_i) = \frac{1}{r_i}\nabla f_i(x_i)$, and $r = [r_1, r_2, \ldots, r_N]^{\mathrm{T}}$ is the left eigenvector of $\mathcal{L}$ associated with the zero eigenvalue. Other notations are the same as the previous section.

Lemma 3: Under Assumptions 1 and 2 , if the communication graph $\mathcal{G}$ is unbalanced, the equilibrium point of (27) is an optimal solution of the distributed optimization problem (2).

Proof: The equilibrium point $(\tilde{x}, \tilde{v})$ of (27) is obtained as

$$\mathbf{0} = \gamma_1[(\hat{\alpha} + \hat{\beta})\mathcal{L} \otimes I_n]\tilde{x}, \tag{28}$$

$$\mathbf{0} = -\gamma_2 \nabla \bar{f}(\tilde{x}) - \gamma_1[(\hat{\alpha} + \hat{\beta})\mathcal{L} \otimes I_n]\tilde{x} - (\mathcal{L} \otimes I_n)\tilde{v}, \tag{29}$$

where $\nabla \bar{f}(x) = [\nabla \bar{f}_1(x_1)^{\mathrm{T}}, \nabla \bar{f}_2(x_2)^{\mathrm{T}}, \ldots, \nabla \bar{f}_N(x_N)^{\mathrm{T}}]^{\mathrm{T}}$. It follows that the equilibrium point satisfies

$$\tilde{x}_i = x^\star, i = 1, 2, \ldots, N, \tag{30}$$

$$\sum_{j=1}^{N} a_{ij}(\tilde{v}_i - \tilde{v}_j) = -\gamma_2 \nabla \bar{f}_i(x^\star), \tag{31}$$

$$\begin{aligned}\gamma_2 \sum_{i=1}^{N} r_i \nabla \bar{f}_i(x^\star) &= \gamma_2 \sum_{i=1}^{N} \nabla f_i(x^\star) \\ &= (r^{\mathrm{T}}\mathcal{L} \otimes \mathbf{1}_n^{\mathrm{T}})\tilde{v} = \mathbf{0}. \end{aligned} \tag{32}$$

Since $\tilde{f}$ is strongly convex, invoking (32), one can obtain that $(\tilde{x}, \tilde{v})$ is a solution of (2) and so is $(\tilde{x}, \tilde{v} + \mathbf{1}_N \otimes \kappa), \kappa \in \mathbb{R}^n$. ■

Theorem 2: Under Assumptions 1 and 2, if the communication graph $\mathcal{G}$ is unbalanced, the dynamic algorithm (27) solves the distributed optimization problem (2) for any $x_i(0), v_i(0) \in \mathbb{R}^n$. Moreover, the dynamic coupling gains α_i will converge to some finite steady-state values.

Proof: The proof is stated in Appendix. ■

Remark 5: Similar to the algorithm (3), the adaptive algorithm (27) guarantees the global stability for the local cost functions f_i with locally Lipschitz gradients. Only the global cost function is assumed to be strongly convex, while the local cost functions f_i can be any differentiable functions.

Remark 6: In the algorithm (27), r is the left eigenvector of $\mathcal{L}$ associated with the zero eigenvalue which implies that the algorithm (27) needs the information of the Laplacian matrix for unbalanced directed graphs.

VI. Simulation Studies

In this section, we show two simulation examples. The first example illustrate the effectiveness of above theoretical results. The second example is an application of our algorithms in solving a regression problem.

A. Example 1

Consider a network of 60 agents whose local cost functions on $\mathbb{R}$ are described by

$$\begin{aligned} f_j =&\sin(x+j), & f_{10+j} =&\cos(\ln(x+j)^2+1)), \\ f_{20+j} =&(x+j)^{\frac{4}{3}}+e^{0.1(x+j)}, & f_{30+j} =&(x+j-4)^4, \\ f_{40+j} =&(x+j+3)^2, & f_{50+j} =&\frac{(x+j)^2}{\sqrt{(x+j)^2+1}}, \end{aligned} \tag{33}$$

for $j = 1,2,\cdots,10$. Note that $f_1, f_2, \cdots, f_{20}$ are periodic functions which are nonconvex. The gradients of $f_{21}, f_{22}, \cdots, f_{40}$ are locally Lipschitz. Moreover, the gradients of $f_{21}, f_{22}, \cdots, f_{30}$ are undifferentiable, i.e., we can not get the Hessians of $f_{21}, f_{22}, \cdots, f_{30}$. Since the global cost function $f(x) = \sum_{i=1}^{60} f_i(x)$ is strongly convex, the global minimizer $x^\star$ is unique.

Two cases of connection graphs are considered for this example. When the connection graph is strongly connected weight-balanced, the adaptive algorithm (3) is applied to solve the distributed optimization problem. The initial states of $x_i(0), v_i(0) \in \mathbb{R}$ are chosen randomly within $[-2, 0.5]$, and the initial values of coupling gains $\alpha_i(0) = 0.01$. For the convergence performance comparisons, the static control gains are chosen as $\gamma_1 = 4, \gamma_2 = 1$ and $\gamma_1 = 4, \gamma_2 = 8$, respectively.

When the connection graph is strongly connected unbalance, the adaptive algorithm (27) is applied. The initial values are the same as that of the weight-balanced case, and the parameters are chosen as $\gamma_1 = 4, \gamma_2 = 1$.

In Figs. 1a(top), 1b(top) and 1c(top), it can be observed that all the trajectories of x_i converge to the global minimizer $x^\star$ (in a black dash-dot line). In Figs. 1a(bottom), 1b(bottom) and 1c(bottom), it can be observed that the dynamic coupling gains α_i converge to some positive steady-state values. From the simulation results in Figs. 1a and 1b we can see that, when γ_2 is increased from 1 to 8, more consensus efforts are needed to deal with the gradients of local objective functions. The dynamic gains α_i in Fig. 1b(bottom) converge to larger positive values than the α_i in Fig. 1a(bottom).

From Figs. 1a and 1c, we can conclude that our adaptive optimization algorithms can solve the distributed convex optimization problem with the unknown network connectivity and the nonconvex local objective functions on both balanced and unbalanced directed graphs.

B. Example 2

In this example, we examine the performance of our proposed algorithms in a practical scenario (e.g., regression problem [16]). Due to the limitation of pages, we only show an example of applying algorithm (3).

The objective of this task is to obtain a predictor of house value by using UCI Housing datasets (available at http://archive.ics.uci.edu/ml /datastes/Housing). Sometimes datasets come from different users, and they do not want to share their private information with others. Hence it is meaningful to employ distributed optimization algorithms.

Consider a network of 6 users interacting over $\mathcal{G}_1$, and each user has 50 datasets. The local cost functions are obtained as

$$f_i(x_i) = \sum_{j=1}^{50} \frac{1}{2}(\nu_j - d_j^{\mathrm{T}} x_i)^2, \forall i = 1,2,\cdots,6,$$

where $x_i \in \mathbb{R}^3$ is the vector of coefficient for linear predictor $\hat{\nu}_j = d_j^{\mathrm{T}} x_i$ is the predicted median monetary value of the house, $\nu_j \in \mathbb{R}$ is the median monetary value of the house, $d_j = [c_j, p_j, 1]^{\mathrm{T}} \in \mathbb{R}^3$, and $c_j, p_j \in \mathbb{R}$ are the per capita crime rate by town and lower status of the population, respectively. The static control gains are chosen as $\gamma_1 = 1, \gamma_2 = 0.2$, and other parameters are chosen in the same way as that of Example 1. Fig. 3a illustrates that the estimated x_i converge to the global optimal value $x^\star \in \mathbb{R}^3$, which is verified by a centralized least squares method. The optimization errors $\|x_i - x^\star\|$ are up bounded by 0.001 after 300 s, and the dynamic coupling gains α_i converge to positive steady-state values. We also emulate the simulation in discrete-time mode, by setting sample time as 0.1 s. Fig. 3b shows that the optimization errors are up bounded by 0.001 after 300 s (3000 iterations). Although the trajectories of $\|x_i - x^\star\|$ and α_i are slightly different in the discrete-time case, the algorithm still guarantee the convergence.

VII. Conclusion

In this note, we have proposed two new adaptive algorithms to solve the distributed optimization problem on directed graphs. By carefully designing adaptive laws, our proposed algorithms achieve global asymptotic convergence when the global cost function is strongly convex and the gradients of local objective functions are locally Lipsthiz. For the strongly connected and weight-balanced graphs, the proposed algorithm is independent of any global information of communication graphs and hence fully distributed. For strongly connected unbalanced graphs, the left eigenvector of the Laplacian matrix associated with the zero eigenvalue is required. Simulation results have illustrated the effectiveness and potential applications of the theoretical results.

Appendix

A. Proof of Theorem 2

Applying the two same state transformations used in Section IV, the network dynamics can be written as

$$\dot{\varrho} =\gamma_1[\mathcal{L}(\hat{\alpha}+\hat{\beta})\otimes I_n]\eta, \tag{34a}$$

$$\begin{aligned}\dot{\eta} =& -\gamma_2(\mathcal{L}\otimes I_n)\bar{h} - \gamma_1[\mathcal{L}(\hat{\alpha}+\hat{\beta})\otimes I_n]\eta \\ & -(\mathcal{L}\otimes I_n)\varrho,\end{aligned} \tag{34b}$$

$$\dot{\alpha}_i =\eta_i^{\mathrm{T}}\eta_i, \tag{34c}$$

where $\bar{h} = \nabla \bar{f}(g+\tilde{x}) - \nabla \bar{f}(\tilde{x})$ and $\beta_i = \eta_i^{\mathrm{T}}\eta_i$.

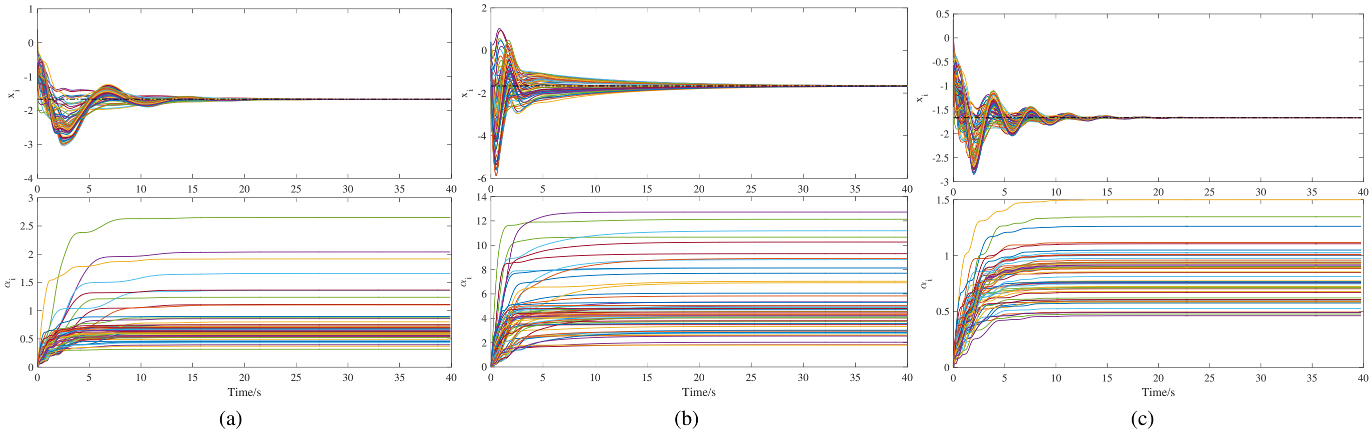

Fig. 1. Simulation results of Example 1; trajectories of the states x_i (top) and the adaptive coupling gains α_i (bottom). (a) weight-balanced graphs case, $\gamma_1 = 4, \gamma_2 = 1$; (b) weight-balanced graphs case, $\gamma_1 = 4, \gamma_2 = 8$; (c) unbalanced graphs case, $\gamma_1 = 4, \gamma_2 = 1$.

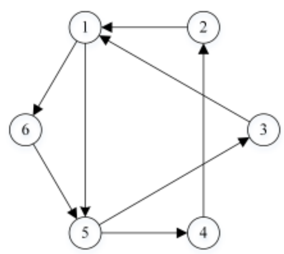

Fig. 2. The strongly connected weight-balanced communication graph $\mathcal{G}_1$.

Let

$$V_4 = \frac{1}{2}\sum_{i=1}^{N} r_i(2\alpha_i + \beta_i)\eta_i^{\mathrm{T}}\eta_i. \tag{35}$$

Following similar analysis in (14)-(20), for $(x, v) \in \mathcal{H}$, it is easy to get the time derivative of $V_1 + V_4$ along the trajectory of (34)

$$\begin{aligned}\dot{V}_1 + \dot{V}_4 \leq & -\frac{\gamma_1\lambda_2(\bar{\mathcal{L}})}{2N}\eta^{\mathrm{T}}[(\hat{\alpha}+\hat{\beta})^2 \otimes I_n]\eta \\ & + \eta^{\mathrm{T}}[(\hat{\alpha} + R\hat{\beta} - \alpha I_N)\otimes I_n]\eta + \frac{4Nr_{\max}^2\lambda_N(\mathcal{L}^{\mathrm{T}}\mathcal{L})}{\gamma_1\lambda_2(\bar{\mathcal{L}})} \\ & \cdot \varrho^{\mathrm{T}}\varrho + \frac{4N\gamma_2^2 r_{\max}^2 M_{\max}^2 \lambda_N(\mathcal{L}^{\mathrm{T}}\mathcal{L})}{\gamma_1 r_{\min}^2 \lambda_2(\bar{\mathcal{L}})} g^{\mathrm{T}} g,\end{aligned} \tag{36}$$

where $R = \mathrm{diag}(r_1, r_2, \ldots, r_N)$, $\bar{\mathcal{L}} = R\mathcal{L} + \mathcal{L}^{\mathrm{T}}R$, $r_{\max} = \max(r_1, r_2, \cdots, r_N)$ and $r_{\min} = \min(r_1, r_2, \cdots, r_N)$.

Consider the following positive definite function

$$V_5 = \frac{1}{2}\sum_{i=1}^{N} r_i(\varrho_i + \eta_i)^{\mathrm{T}}(\varrho_i + \eta_i). \tag{37}$$

The time derivative of (37) is described by

$$\begin{aligned}\dot{V}_5 = & \sum_{i=1}^{N} r_i(\varrho_i^{\mathrm{T}}\dot{\varrho}_i + \eta_i^{\mathrm{T}}\dot{\eta}_i + \varrho_i^{\mathrm{T}}\dot{\eta}_i + \eta_i^{\mathrm{T}}\dot{\varrho}_i) \\ \leq & \frac{\lambda_2(\bar{\mathcal{L}}) + 2r_{\max}^2\lambda_N(\mathcal{L}^{\mathrm{T}}\mathcal{L})}{\lambda_2(\bar{\mathcal{L}})}\eta^{\mathrm{T}}\eta - \frac{\lambda_2(\bar{\mathcal{L}})}{4}\varrho^{\mathrm{T}}\varrho \\ & + \frac{\gamma_2^2 r_{\max}^2 M_{\max}^2\lambda_N(\mathcal{L}^{\mathrm{T}}\mathcal{L})(\lambda_2(\bar{\mathcal{L}})+8)}{4r_{\min}^2\lambda_2(\bar{\mathcal{L}})} g^{\mathrm{T}} g,\end{aligned} \tag{38}$$

where we have used Lemma 1 and the fact $\varrho^{\mathrm{T}}(R\mathcal{L}\otimes I_n)\varrho = \frac{\varrho^{\mathrm{T}}(\bar{\mathcal{L}}\otimes I_n)\varrho}{2}$.

A Lyapunov function candidate for the whole closed-loop system is chosen as

$$\bar{V} = V_1 + V_4 + \frac{17Nr_{\max}^2\lambda_N(\mathcal{L}^{\mathrm{T}}\mathcal{L})}{\gamma_1\lambda_2(\bar{\mathcal{L}})^2}V_5. \tag{39}$$

Applying the results (36) and (38), and from (39), we can obtain

$$\begin{aligned}\dot{\bar{V}} \leq & \eta^{\mathrm{T}}\Bigg[\Bigg(-\frac{\gamma_1\lambda_2(\bar{\mathcal{L}})}{2N}(\hat{\alpha}+\hat{\beta})^2 + (\hat{\alpha} + R\hat{\beta}) - \alpha I_N \\ & + \frac{17Nr_{\max}^2\lambda_N(\mathcal{L}^{\mathrm{T}}\mathcal{L})(\lambda_2(\bar{\mathcal{L}}) + 2r_{\max}^2\lambda_N(\mathcal{L}^{\mathrm{T}}\mathcal{L}))}{\gamma_1\lambda_2(\bar{\mathcal{L}})^3}I_N\Bigg) \\ & \otimes I_n\Bigg]\eta + \Bigg(\frac{4N\gamma_2^2 r_{\max}^2 M_{\max}^2\lambda_N(\mathcal{L}^{\mathrm{T}}\mathcal{L})}{\gamma_1 r_{\min}^2\lambda_2(\bar{\mathcal{L}})} \\ & + \frac{17N\gamma_2^2 r_{\max}^4 M_{\max}^2\lambda_N(\mathcal{L}^{\mathrm{T}}\mathcal{L})^2(\lambda_2(\bar{\mathcal{L}})+8)}{4\gamma_1 r_{\min}^2\lambda_2(\bar{\mathcal{L}})^3}\Bigg) g^{\mathrm{T}} g \\ & - \frac{Nr_{\max}^2\lambda_N(\mathcal{L}^{\mathrm{T}}\mathcal{L})}{4\gamma_1\lambda_2(\bar{\mathcal{L}})}\varrho^{\mathrm{T}}\varrho.\end{aligned} \tag{40}$$

By Young's inequality, we have

$$\begin{aligned}\eta^{\mathrm{T}}(\hat{\alpha} + R\hat{\beta})\eta \leq & \eta^{\mathrm{T}}\Bigg[\Bigg(\frac{\gamma_1\lambda_2(\bar{\mathcal{L}})}{4N}\beta^2 + \frac{\gamma_1\lambda_2(\bar{\mathcal{L}})}{4N}\alpha^2 \\ & + \frac{N(r_{\max}^2+1)}{\gamma_1\lambda_2(\bar{\mathcal{L}})}I_N\Bigg)\otimes I_n\Bigg]\eta.\end{aligned} \tag{41}$$

Let $\bar{\delta} \in \mathbb{R}_{>0}$ be an arbitrary small positive constant. Since $g(x(t))^{\mathrm{T}}g(x(t)) \leq g(x(0))^{\mathrm{T}}g(x(0))$, for $(x, v) \in \mathcal{H}$, when $\eta^{\mathrm{T}}\eta \geq \bar{\delta}$, there always exists a sufficiently large positive

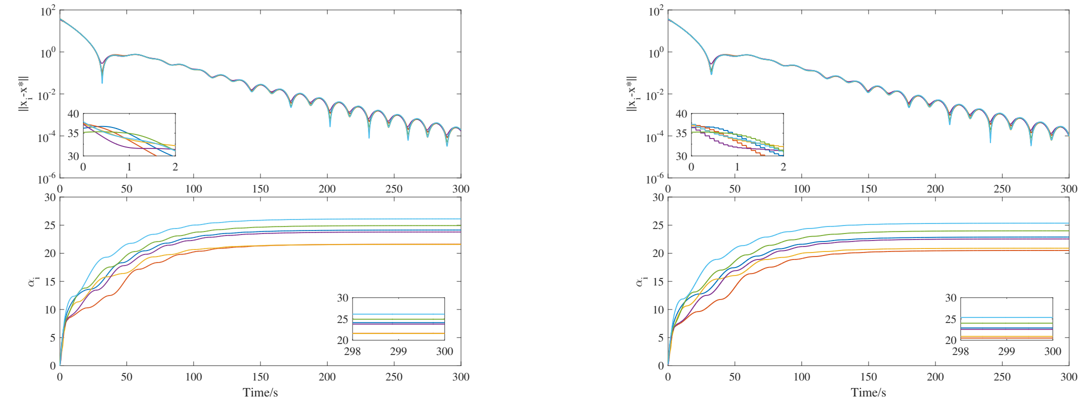

Fig. 3. Simulation results of Example 2. (a) The estimated errors $\|x_i - x^\star\|$ (top) and the adaptive coupling gains α_i (bottom) of Example 2 in continuous-time mode; (b) the estimated errors $\|x_i - x^\star\|$ (top) and the adaptive coupling gains α_i (bottom) in discrete-time mode.

scalar $\bar{\epsilon} \in \mathbb{R}_{>0}$ such that $\bar{\epsilon}\eta^{\mathrm{T}}\eta \geq \Big(\frac{4N\gamma_2^2 r_{\max}^2 M_{\max}^2 \lambda_N(\mathcal{L}^{\mathrm{T}}\mathcal{L})}{\gamma_1 r_{\min}^2 \lambda_2(\bar{\mathcal{L}})} + \frac{17N\gamma_2^2 r_{\max}^4 M_{\max}^2 \lambda_N(\mathcal{L}^{\mathrm{T}}\mathcal{L})^2(\lambda_2(\bar{\mathcal{L}})+8)}{4\gamma_1 r_{\min}^2 \lambda_2(\bar{\mathcal{L}})^3}\Big) g^{\mathrm{T}} g$.

By incorporating this fact and (41) into (40), and choosing $\alpha \geq \bar{\epsilon} + \frac{17Nr_{\max}^2\lambda_N(\mathcal{L}^{\mathrm{T}}\mathcal{L})(\lambda_2(\bar{\mathcal{L}})+2r_{\max}^2\lambda_N(\mathcal{L}^{\mathrm{T}}\mathcal{L}))}{\gamma_1\lambda_2(\bar{\mathcal{L}})^3} + \frac{N(r_{\max}^2+1)}{\gamma_1\lambda_2(\bar{\mathcal{L}})} + \frac{4N}{\gamma_1\lambda_2(\bar{\mathcal{L}})}$, we have

$$\begin{aligned}\dot{\hat{V}} \leq & \eta^{\mathrm{T}}\left[\left(-\frac{\gamma_1\lambda_2(\bar{\mathcal{L}})}{4N}(\hat{\alpha}+\hat{\beta})^2 - \frac{4N}{\gamma_1\lambda_2(\bar{\mathcal{L}})} I_n\right)\otimes I_N\right]\eta \\ & - \frac{Nr_{\max}^2\lambda_N(\mathcal{L}^{\mathrm{T}}\mathcal{L})}{4\gamma_1\lambda_2(\bar{\mathcal{L}})}\varrho^{\mathrm{T}}\varrho \\ \leq & -\eta^{\mathrm{T}}(\hat{\alpha}(0)\otimes I_n)\eta - \frac{Nr_{\max}^2\lambda_N(\mathcal{L}^{\mathrm{T}}\mathcal{L})}{4\gamma_1\lambda_2(\bar{\mathcal{L}})}\varrho^{\mathrm{T}}\varrho, \end{aligned} \quad (42)$$

where we have used the fact that α_i are monotonically increasing and $\alpha_i(0) \in \mathbb{R}_{>0}$. The rest of proof follows similarly as that of Theorem 1.

REFERENCES

[1] X. Nguyen, M. I. Jordan, and B. Sinopoli, "A kernel-based learning approach to ad hoc sensor network localization," *ACM Trans. Sen. Netw.*, vol. 1, no. 1, pp. 134–152, Aug. 2005.

[2] S. S. Ram, V. V. Veeravalli, and A. Nedic, "Distributed and recursive parameter estimation in parametrized linear state-space models," *IEEE Trans. Autom. Control*, vol. 55, no. 2, pp. 488–492, Feb 2010.

[3] A. Cherukuri and J. Corts, "Distributed generator coordination for initialization and anytime optimization in economic dispatch," *IEEE Trans. Control Netw. Syst.*, vol. 2, no. 3, pp. 226–237, Sept 2015.

[4] S. S. Ram, A. Nedic, and V. V. Veeravalli, "A new class of distributed optimization algorithms: application to regression of distributed data," *Optimiz. Methods Software*, vol. 27, no. 1, pp. 71–88, 2012.

[5] B. Johansson, M. Rabi, and M. Johansson, "A randomized incremental subgradient method for distributed optimization in networked systems," *SIAM J. Optimiz.*, vol. 20, no. 3, pp. 1157–1170, 2010.

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

[7] J. Koshal, A. Nedic, and U. V. Shanbhag, "Multiuser optimization: Distributed algorithms and error analysis," *SIAM J. Optimiz.*, vol. 21, no. 3, pp. 1046–1081, 2011.

[8] I. Lobel and A. Ozdaglar, "Distributed subgradient methods for convex optimization over random networks," *IEEE Trans. Autom. Control*, vol. 56, no. 6, pp. 1291–1306, June 2011.

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

[10] M. Zhu and S. Martinez, "On distributed convex optimization under inequality and equality constraints," *IEEE Trans. Autom. Control*, vol. 57, no. 1, pp. 151–164, Jan 2012.

[11] J. Wang and N. Elia, "A control perspective for centralized and distributed convex optimization," in *Proc. 50th IEEE Conf. Decision Control Eur. Control Conf.*, Dec 2011, pp. 3800–3805.

[12] B. Gharesifard and J. Corts, "Distributed continuous-time convex optimization on weight-balanced digraphs," *IEEE Trans. Autom. Control*, vol. 59, no. 3, pp. 781–786, March 2014.

[13] S. S. Kia, J. Corts, and S. Martnez, "Distributed convex optimization via continuous-time coordination algorithms with discrete-time communication," *Automatica*, vol. 55, pp. 254–264, 2015.

[14] J. Lu and C. Y. Tang, "Zero-gradient-sum algorithms for distributed convex optimization: The continuous-time case," *IEEE Trans. Autom. Control*, vol. 57, no. 9, pp. 2348–2354, Sept 2012.

[15] Q. Liu and J. Wang, "A second-order multi-agent network for bound-constrained distributed optimization," *IEEE Trans. Autom. Control*, vol. 60, no. 12, pp. 3310–3315, Dec 2015.

[16] D. Varagnolo, F. Zanella, A. Cenedese, G. Pillonetto, and L. Schenato, "Newton-raphson consensus for distributed convex optimization," *IEEE Trans. Autom. Control*, vol. 61, no. 4, pp. 994–1009, April 2016.

[17] S. Lee, A. Ribeiro, and M. M. Zavlanos, "Distributed continuous-time online optimization using saddle-point methods," in *Proc. 55th IEEE Conf. Decision Control*, Dec 2016, pp. 4314–4319.

[18] S. Yang, Q. Liu, and J. Wang, "Distributed optimization based on a multiagent system in the presence of communication delays," *IEEE Trans. Syst., Man, Cybern., Syst.*, vol. PP, no. 99, pp. 1–12, 2016.

[19] Y. Song and W. Chen, "Finite-time convergent distributed consensus optimisation over networks," *IET Control Theory Appl.*, vol. 10, no. 11, pp. 1314–1318, 2016.

[20] Y. Zhao, Y. Liu, G. Wen, and G. Chen, "Distributed optimization of linear multi-agent systems: Edge- and node-based adaptive designs," *IEEE Trans. Autom. Control*, vol. PP, no. 99, pp. 1–1, 2017.

[21] P. Bianchi and J. Jakubowicz, "Convergence of a multi-agent projected stochastic gradient algorithm for non-convex optimization," *IEEE Trans. Autom. Control*, vol. 58, no. 2, pp. 391–405, Feb 2013.

[22] B. Touri and B. Gharesifard, "Continuous-time distributed convex optimization on time-varying directed networks," in *Proc. 54th IEEE Conf. Decision Control*, Dec 2015, pp. 724–729.

[23] Z. Li, G. Wen, Z. Duan, and W. Ren, "Designing fully distributed consensus protocols for linear multi-agent systems with directed graphs," *IEEE Trans. Autom. Control*, vol. 60, no. 4, pp. 1152–1157, April 2015.

[24] Z. Li and Z. Ding, "Distributed adaptive consensus and output tracking of unknown linear systems on directed graphs," *Automatica*, vol. 55, pp. 12–18, 2015.

[25] Z. Ding, "Consensus disturbance rejection with disturbance observers," *IEEE Trans. Ind. Electron.*, vol. 62, no. 9, pp. 5829–5837, Sept 2015.

[26] Z. Li and Z. Duan, *Cooperative Control of Multi-Agent Systems: A Consensus Region Approach*. CRC Press, 2014.