2017 IEEE 56th Annual Conference on Decision and Control (CDC)
December 12-15, 2017, Melbourne, Australia

Network Flows as Least Squares Solvers for Linear Equations

Yang Liu, Youcheng Lou, Brian D. O. Anderson and Guodong Shi

Abstract— **This paper presents a first-order continuous-time distributed step-size algorithm for computing the least squares solution to a linear equation over networks. Given the uniqueness of the solution and nonintegrable step size, the convergence results are provided for fixed graphs. For the nonunique solution and square integrable step size, the convergence is shown for constantly connected switching graphs. We also validate the results and illustrate possible impacts on the convergence speed using a few numerical examples.**

I. Introduction

In modern engineering systems, there is a great demand for large-scale computing capabilities of solving real-world mathematical problems. Centralized algorithms are effective tools if the computing center possesses the information of the entire problem. In some cases, however, due to the comparatively weak computing power of one agent and its limited knowledge of the whole problem, the notion of distributed computation over networks is developed, which is nowadays widely applied in the areas of analyzing the consensus of complex systems [1], solving various optimization problems [2], carrying out distributed estimation [3], [4] and filtering [5].

Solving systems of linear equations using distributed algorithms over networks, which is intrinsically related to distributed optimization problems, emerges as one of the basic tasks in distributed computation. In these scenarios, it is often assumed that each agent of the network only has access to one or a few of the linear equations due to security issues or memory limitation, and is only permitted to interact with some of the other agents. Each agent of the network may request the entire solution, instead of only its components, for the purpose of protecting the privacy of customers [6]. On the one hand, a number of contributions have been made to the development of distributed solvable linear equation solvers, where simple first-order distributed algorithms, in continuous time or discrete time [7]–[14], manage to deliver satisfactory solutions even for switching network structures. On the other hand, another frequent case in practical problems is concerning with non-solvable linear equations, in which we often consider least squares solution by minimizing the associated objective function.

However, it seems a rather challenging problem in developing distributed least squares solvers for network linear equations, due to the mismatch between individual linear equations at each node and the network least squares solution. Despite the difficulties, there exist a few distributed algorithms developed for the least squares problem using different approaches, such as second-order algorithms [15]–[18], state expansion [12] and high gain consensus gain method [8]. Second-order distributed least squares solvers [15]–[18] generally can produce good convergence performance, however, they rely on restricted network structures and demand higher communication and storage capacities. State expansion method [12] is to enlarge the state dimension and then apply the existing methods for linear equations with exact solutions directly, but the nodes have to grasp more knowledge than their own linear equations. It was shown in [8] that first-order algorithms for exact solutions can be adapted to the least squares case by a high consensus gain, but only in approximate sense.

In this paper, we propose a first-order continuous-time flow for the least squares problems of network linear equations, in which each agent keeps averaging the state with its neighbors' and in the mean time descending along the diminishing gradient of its local cost function. This flow is inspired by the work of [19] on distributed subgradient optimization. If the network linear equation has one unique least squares solution, we prove that all node states asymptotically converge to that solution along our flow, with constant and connected graphs and nonintegrable step size. For switching network structure that is kept connected, we show that the node states always converge to one of the least squares solutions with square integrable step size. We also provide a few numerical examples that validate the usefulness of the proposed algorithms and demonstrate the convergence rate.

The remainder of this paper is organized as follows. In Section II, a brief introduction to the definition of the problem studied is given. We present the main results in Section III and make further discussions using numerical examples in Section IV. In Section V, the main work of this paper is summarized and potential future work are provided.

II. Problem Definition

In this section, a few mathematical preliminaries are provided, regarding linear equations over networks. Also we establish a distributed network flow that can asymptotically compute the least squares solution to network linear equations and discuss its relation to existing work.

This work was supported by the Australian Research Council (ARC) under grants DP-130103610 and DP-160104500.

Y. Liu is with Research School of Engineering, The Australian National University, Canberra 0200, Australia yang.liu@anu.edu.au

Y. Lou is with the Department of Systems Engineering and Engineering Management, The Chinese University of Hong Kong, Hong Kong louyoucheng@amss.ac.cn

B. D. O. Anderson is with Research School of Engineering, The Australian National University, Canberra 0200, Australia brian.anderson@anu.edu.au

G. Shi is with Research School of Engineering, The Australian National University, Canberra 0200, Australia guodong.shi@anu.edu.au

978-1-5090-2872-6/17/$31.00 ©2017 IEEE

A. Linear Equations

Consider the following linear algebraic equation with respect to $\mathbf{y} \in \mathbb{R}^m$:

$$\mathbf{z} = \mathbf{H}\mathbf{y} \tag{1}$$

where $\mathbf{z} \in \mathbb{R}^N$ and $\mathbf{H} \in \mathbb{R}^{N\times m}$ are known ($m \le N$). Denote

$$\mathbf{H} = \begin{bmatrix} \mathbf{h}_1^\top \\ \mathbf{h}_2^\top \\ \vdots \\ \mathbf{h}_N^\top \end{bmatrix}, \quad \mathbf{z} = \begin{bmatrix} z_1 \\ z_2 \\ \vdots \\ z_N \end{bmatrix}.$$

We can rewrite (1) as

$$\mathbf{h}_i^\top \mathbf{y} = z_i, \ i = 1, \dots, N.$$

Denote the column space of a matrix $\mathbf{M}$ by $\mathrm{colsp}\{\mathbf{M}\}$. If $\mathbf{z} \in \mathrm{colsp}\{\mathbf{H}\}$, then the equation (1) always has (one or many) exact solutions. If $\mathbf{z} \notin \mathrm{colsp}\{\mathbf{H}\}$, the least squares solution is defined by the solution of the following optimization problem:

$$\min_{\mathbf{y}\in\mathbb{R}^m} \|\mathbf{z} - \mathbf{H}\mathbf{y}\|^2. \tag{2}$$

It is well known that if $\mathrm{rank}(\mathbf{H}) = m$, then (2) yields a unique solution $\mathbf{y}^* = (\mathbf{H}^\top\mathbf{H})^{-1}\mathbf{H}^\top\mathbf{z}$, while (2) has a set of non-unique least squares solutions if $\mathrm{rank}(\mathbf{H}) < m$. Define a cost function $f(\mathbf{y}) = \sum_{i=1}^{N} f_i(\mathbf{y})$ where $f_i(\mathbf{y}) = |\mathbf{h}_i^\top\mathbf{y} - z_i|^2$. Note that $\mathbf{y}^* \in \mathrm{argmin}\, f(\mathbf{y})$, i.e., $\nabla f(\mathbf{y}^*) = 0$, where $\nabla f(\mathbf{y}) = 2\sum_{i=1}^{N}(\mathbf{h}_i\mathbf{h}_i^\top\mathbf{y} - z_i\mathbf{h}_i)$.

B. Networks

Denote $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ as a constant undirected graph with the finite set of nodes $\mathcal{V} = \{1, 2, \dots, N\}$ and the set of edges $\mathcal{E} \subset \mathcal{V}\times\mathcal{V}$. Suppose throughout the rest of paper that $\mathcal{G}$ has no self loops. Assign a weight function $w : \mathcal{E} \to \mathbb{R}$ to every edge. Obviously $w((i,j)) = w((j,i))$. Based on constant graphs, we next introduce time-varying graphs. Let $\mathcal{Q}$ be the set containing all possible constant and undirected graphs induced by the node set $\mathcal{V}$ and $\mathcal{Q}^* \subset \mathcal{Q}$ be a subset of $\mathcal{Q}$. Define a piecewise constant mapping $\mathcal{G}_{\sigma(\cdot)} = (\mathcal{V}, \mathcal{E}_{\sigma(\cdot)}) : \mathbb{R}^{\ge 0} \to \mathcal{Q}^*$. Note that the time-varying graph $\mathcal{G}_{\sigma(t)} = (\mathcal{V}, \mathcal{E}_{\sigma(t)})$ represents the network topology at time t. Let $\mathcal{N}_i(t)$ be the set of neighbor nodes that are connected to node i at time t, i.e., $\mathcal{N}_i(t) = \{j : (i,j) \in \mathcal{E}_{\sigma(t)}\}$. Define the degree matrix $\mathbf{D}(t) = \mathrm{diag}(|\mathcal{N}_1(t)|, |\mathcal{N}_2(t)|, \dots, |\mathcal{N}_N(t)|)$ and the adjacency matrix $\mathbf{A}(t)$ of the graph $\mathcal{G}_{\sigma(t)}$ by $[\mathbf{A}(t)]_{ij} = w((i,j))$ if $(i,j) \in \mathcal{E}_{\sigma(t)}\}$, and $[\mathbf{A}(t)]_{ij} = 0$ otherwise. Then $\mathbf{L}(t) = \mathbf{D}(t) - \mathbf{A}(t)$ is the Laplacian of graph $\mathcal{G}_{\sigma(t)}$ at time t.

C. Distributed Flows

Assume that node i of the network $\mathcal{G}_{\sigma(t)}$ only knows the information of $\mathbf{h}_i, z_i$, i.e., node i is associated with the linear equation $\mathbf{h}_i^\top\mathbf{y} = z_i$. We assign $\mathbf{x}_i(t) \in \mathbb{R}^m$ that varies as a function of time t as the state of each node i. Then we propose the following continuous-time network flow

$$\dot{\mathbf{x}}_i(t) = K\sum_{j\in\mathcal{N}_i(t)} [\mathbf{A}(t)]_{ij}(\mathbf{x}_j(t) - \mathbf{x}_i(t)) - \frac{\alpha(t)}{2}\nabla f_i(\mathbf{x}_i(t)), \tag{3}$$

where $K \in \mathbb{R}^+$ is a positive constant, $\nabla f_i(\mathbf{y}) = 2(\mathbf{h}_i\mathbf{h}_i^\top\mathbf{y} - z_i\mathbf{h}_i)$ and the step size $\alpha(t) : \mathbb{R}^{\ge 0} \to \mathbb{R}^+$ is a continuous function which assures the continuity of all $\mathbf{x}_i(t)$ and their derivatives, with the exception of the time points when the networks switch. In vector form, we have

$$\dot{\mathbf{x}}(t) = -\mathbf{M}(t)\mathbf{x}(t) + \alpha(t)\mathbf{z}_H \tag{4}$$

where $\mathbf{M}(t) = K(\mathbf{L}(t) \otimes \mathbf{I}_m) + \alpha(t)\tilde{\mathbf{H}}$, $\mathbf{x}(t) = [\mathbf{x}_1^\top(t) \ \dots \ \mathbf{x}_N^\top(t)]^\top \in \mathbb{R}^{Nm}$, $\tilde{\mathbf{H}} = \mathrm{diag}\{\mathbf{h}_1\mathbf{h}_1^\top, \dots, \mathbf{h}_N\mathbf{h}_N^\top\} \in \mathbb{R}^{Nm\times Nm}$, and $\mathbf{z}_H = [z_1\mathbf{h}_1^\top \ \dots \ z_N\mathbf{h}_N^\top]^\top \in \mathbb{R}^{Nm}$. Now we make a few assumptions of $\alpha(t)$ that will be used in our main results.

Assumption 1: (i) $\int_0^\infty \alpha(t)dt = \infty$; (ii) $\lim_{t\to\infty}\alpha(t) = 0$; (iii) $\int_0^\infty \alpha^2(t)dt < \infty$.

D. Discussion

Now we clarify the relation between the previous work of distributed least squares and optimization algorithms, and our algorithm (3) by briefly discussing their structure and applicability. It is clear that (3) has exactly the same structure as the flow in [19], [20], with the difference that the flow in [19], [20] is discrete time but (3) is continuous time. However, we cannot use the algorithm and the analysis directly because the gradient boundedness of (3) is not directly verifiable. It can be noted that the first-order flow in [8] is a special case of (3) by letting $\alpha(t)$ be a proper constant. Due to the existence of the diminishing step size, (3) is a linear time-varying system, while the flow in [8] is linear time invariant and can only produce the solution in approximate sense. Hence the approach to analyzing the flow in [8] is not applicable for (3). There are also second-order least squares solvers [15]–[18], nevertheless they often request limited network topologies and have more complex structures than (3).

III. Convergence Results

In this section, we investigate the flow (4) over fixed and switching networks, respectively, and establish the convergence conditions regarding $\alpha(t)$ and the graphs.

A. Convergence over Fixed Networks

First we consider the case where the linear equation (1) has one unique least squares solution and the network is a constant graph for all t. In this case, the following theorem can be proved.

Theorem 1: Suppose $\mathrm{rank}(\mathbf{H}) = m$ and denote $\mathbf{y}^* = (\mathbf{H}^\top\mathbf{H})^{-1}\mathbf{H}^\top\mathbf{z}$ as the unique least squares solution of (1). Let Assumption 1.(i) and (ii) hold. If $\mathcal{G}_{\sigma(t)} = \mathcal{G}$ is constant and connected for all $t \ge 0$, then along any solution of (3) there holds that

$$\lim_{t\to\infty}\mathbf{x}_i(t) = \mathbf{y}^*$$

for all $i \in \mathcal{V}$.

B. Convergence over Switching Networks

Now we consider a more general case where the least squares solutions of (1) can be unique or non-unique, and the network $\mathcal{G}_{\sigma(t)}$ switches among a collection of graphs. Regarding the convergence of (3) in this case, we have the following theorem.

Theorem 2: Suppose $\mathrm{rank}(\mathbf{H}) \le m$ and denote the set of least squares solutions of (1) by $\mathcal{Y}_{\mathrm{LS}} = \mathrm{argmin}\, f(\mathbf{y})$. In particular, $|\mathcal{Y}_{\mathrm{LS}}| = 1$ if $\mathrm{rank}(\mathbf{H}) = m$. Let Assumption 1.(i), (ii) and (iii) hold. If all $\mathcal{G} \in \mathcal{Q}^*$ are connected, then along any solution of (3) over the switching graph $\mathcal{G}_{\sigma(t)}$ there exists $\hat{\mathbf{y}} \in \mathcal{Y}_{\mathrm{LS}}$ such that

$$\lim_{t\to\infty} \mathbf{x}_i(t) = \hat{\mathbf{y}}$$

for all $i \in \mathcal{V}$.

We conjecture that the condition for graphs in Theorem 2 can be even more relaxed. Now we define the graph union of $\mathcal{Q}^*$ as $\mathcal{G}(\mathcal{Q}^*) = \bigcup_{\mathcal{G}\in\mathcal{Q}^*} \mathcal{E}$ with $\mathcal{G} = (\mathcal{V}, \mathcal{E})$. One of the potential generalizations is to extend "all graphs $\mathcal{G} \in \mathcal{Q}^*$ are connected" to "the graph union $\mathcal{G}(\mathcal{Q}^*)$ is connected", for which we provide a numerical example in this paper.

C. Proofs of Statements

Now we provide the proofs of Theorem 1 and Theorem 2, in addition to a couple of key lemmas.

1) Key Lemmas: Here are a few lemmas that assist with the proofs of Theorem 1 and Theorem 2. The proofs of Lemma 1 and Lemma 2 are directly based on properties of strongly convex functions and Grönwall's Inequality, respectively, and omitted due to space limitations.

Lemma 1: Consider a linear equation $\mathbf{H}\mathbf{y} = \mathbf{z}$ with respect to $\mathbf{y} \in \mathbb{R}^m$ where $\mathbf{H} \in \mathbb{R}^{N\times m}, \mathbf{z} \in \mathbb{R}^N$. Denote $\mathbf{h}_i^\top$ as the i-th row of $\mathbf{H}$ and z_i as the i-th entry of $\mathbf{z}$. Let $f(\mathbf{y}) = \sum_{i=1}^N |\mathbf{h}_i^\top \mathbf{y} - z_i|^2$. If $\mathrm{rank}(\mathbf{H}) = m$, then $f(\mathbf{y})$ is strongly convex and there exists $\sigma > 0$ such that $f(\mathbf{y}) \ge f(\mathbf{y}^*) + \frac{\sigma}{2}\|\mathbf{y} - \mathbf{y}^*\|^2$ with $\mathbf{y}^* = \min_{\mathbf{y}} f(\mathbf{y})$.

Lemma 2: Consider a continuously differentiable function $g(t) : \mathbb{R}^{\ge 0} \to \mathbb{R}^{\ge 0}$ with $g(0) \ge 0$. If there exist continuous functions $\gamma(t) : \mathbb{R}^{\ge 0} \to \mathbb{R}^+$ and $\beta(t) : \mathbb{R}^{\ge 0} \to \mathbb{R}^+$ satisfying $\dot{g}(t) \le -\gamma(t)g(t) + \beta(t)$, then $g(t) \le \exp(-\int_0^t \gamma(s)ds)g(0) + \int_0^t \exp(-\int_s^t \gamma(r)dr)\beta(s)ds$. Furthermore, the following statements hold:

(i) If $\int_0^\infty \gamma(t)dt = \infty$ and $\lim_{t\to\infty} \frac{\beta(t)}{\gamma(t)} = 0$, then $\lim_{t\to\infty} g(t) = 0$.

(ii) If $\int_0^\infty \gamma(t)dt = \infty$ and $\limsup_{t\to\infty} \frac{\beta(t)}{\gamma(t)} < \infty$, then $g(t)$ is bounded.

2) Proof of Theorem 1: The proof starts by establishing $\mathbf{x}(t)$ is bounded, which is given as follows. Consider

$$\begin{aligned} Q_K(\mathbf{x}, t) &= \mathbf{x}^\top \mathbf{M}(t)\mathbf{x} \\ &= K \sum_{\{i,j\}\in\mathcal{E}} [\mathbf{A}]_{ij}\|\mathbf{x}_j - \mathbf{x}_i\|^2 + \alpha(t)\sum_{i=1}^N |\mathbf{h}_i^\top \mathbf{x}_i|^2 \end{aligned}$$

with $\mathbf{x} \ne 0$. It can be easily known that $Q_K(\mathbf{x}, t) \ge 0$ and the equality holds only if $\mathbf{x}_i = \mathbf{x}_j$ for any i, j. If $\mathrm{rank}(\mathbf{H}) = m$, there does not exist $\mathbf{x} \ne 0$ such that $Q_K(\mathbf{x}) = 0$, i.e., $Q_K(\mathbf{x}) > 0$ for $\mathbf{x} \ne 0$. Therefore, $\mathbf{M}(t)$ is positive-definite for all t. Similarly, $\mathbf{P} = \mathbf{L} \otimes \mathbf{I}_m + \tilde{\mathbf{H}}$ is also positive-definite. Under Assumption 1.(ii), we know that there exists sufficiently large t_0 such that $\alpha(t) < K$ for all $t > t_0$. By Theorem 4.2.2 in [21], we know that $Q_K(\mathbf{x}, t) \ge \alpha(t)\mathbf{x}^\top \mathbf{P}\mathbf{x} \ge \alpha(t)\lambda_{min,P}\|\mathbf{x}\|^2$ for any $\mathbf{x}$ and all $t > t_0$, where $\lambda_{min,P}$ is the minimum eigenvalue of $\mathbf{P}$. Let $h(t) = \|\mathbf{x}(t)\|^2$. Then

$$\begin{aligned} \frac{d}{dt}h(t) &= -2\mathbf{x}^\top(K(\mathbf{L}\otimes\mathbf{I}_m) + \alpha(t)\tilde{\mathbf{H}})\mathbf{x} + 2\alpha(t)\mathbf{x}^\top \mathbf{z}_H \\ &\le -2\alpha(t)\lambda_{min,P}\|\mathbf{x}\|^2 + 2\alpha(t)\|\mathbf{x}\|\|\mathbf{z}_H\|, \end{aligned}$$

for $t > t_0$. Consider

$$\frac{d}{dt}\sqrt{h} = \frac{\dot{h}}{2\sqrt{h}} \le -\alpha(t)\lambda_{min,P}\sqrt{h} + \alpha(t)\|\mathbf{z}_H\|,\ t \ge t_0.$$

By Lemma 2.(ii), identifying $g(t)$ with $\sqrt{h(t)}$, we have that $\sqrt{h(t)} = \|\mathbf{x}(t)\|$ is bounded for $t > t_0$. Due to the continuity of $\mathbf{x}(t)$, $\|\mathbf{x}(t)\|$ is bounded for all $t \ge 0$.

Denote $\bar{\mathbf{x}}(t) := \frac{1}{N}\sum_{i=1}^N \mathbf{x}_i(t)$ and $\bar{\mathbf{x}}^\diamond(t) := \mathbf{1}_N \otimes \bar{\mathbf{x}}(t)$. Then we analyze $\frac{d}{dt}\|\mathbf{x}(t) - \bar{\mathbf{x}}^\diamond(t)\|^2$ and find out

$$\frac{d}{dt}\|\mathbf{x}(t) - \bar{\mathbf{x}}^\diamond(t)\|^2 \le -2\lambda_2 K\|\mathbf{x}(t) - \bar{\mathbf{x}}^\diamond(t)\|^2 + \beta(t),$$

where $\beta(t) = 2\alpha(t)\langle \mathbf{x}(t) - \bar{\mathbf{x}}^\diamond(t), \mathbf{z}_H - \tilde{\mathbf{H}}\mathbf{x}(t) - \mathbf{1}_N \otimes (\frac{1}{2N}\sum_{i=1}^N \nabla f_i(\mathbf{x}_i))\rangle$ and λ_2 is the second minimum eigenvalue of $\mathbf{L}$. Under Assumption 1.(ii) and by the claim that $\|\mathbf{x}(t)\|$ is bounded, we know that $\lim_{t\to\infty}\beta(t) = 0$. By Lemma 2.(i), $\lim_{t\to\infty}\|\mathbf{x}(t) - \bar{\mathbf{x}}^\diamond(t)\|^2 = 0$, i.e., the dynamical system (4) achieves a consensus. Similarly, we study $\frac{d}{dt}\|\bar{\mathbf{x}}(t) - \mathbf{y}^*\|^2$ and apply Lemma 1, Lemma 2.(i). It can be easily obtained that $\lim_{t\to\infty}\|\bar{\mathbf{x}}(t) - \mathbf{y}^*\|^2 = 0$. Thus one can conclude that (4) reaches a consensus that is the least squares solution to (1).

3) Proof of Theorem 2: The idea of proving Theorem 2 is analogous to the proof of Theorem 1, and omitted due to page limit.

IV. Numerical Examples

In this section, a couple of numerical examples are provided to validate the results of Theorem 1, 2.

A. Fixed Graphs

Example 1. Consider a linear algebraic equation with respect to $\mathbf{y} \in \mathbb{R}^2$:

$$\begin{bmatrix} -1 & 1 \\ 2 & 1 \\ 2 & 0 \\ 1 & 0 \end{bmatrix} \mathbf{y} = \begin{bmatrix} 2 \\ 1 \\ 0 \\ 2 \end{bmatrix}.$$

It can be calculated that the unique least squares solution is $\mathbf{y}^* = [0.0526\ 1.4737]^\top$. Given the initial value $\mathbf{x}(0) = [0.3\ -0.2\ 0.1\ -0.1\ 1.1\ 1.6\ -0.2\ 0.8]^\top$ and $K = 100$. Let the network flow (4) do iteration over the graph $\mathcal{G}_0$ given in Figure 1 for $\alpha(t) = t^{-1}, \alpha(t) = t^{-\frac{1}{2}}, \alpha(t) = t^{-\frac{1}{4}}, \alpha(t) =$

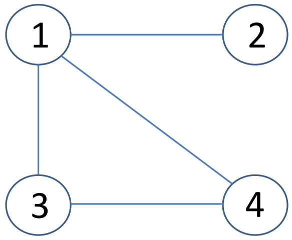


Fig. 1. A constant, connected and undirected graph $\mathcal{G}_0$ considered in Example 1 and 2.

$t^{-\frac{1}{8}}$, respectively, in which cases the conditions of Theorem 1 are satisfied. Then we plot the trajectories of $e(t) = \|\mathbf{x}(t) - \mathbf{1}_4 \otimes \mathbf{y}^*\|$ in logarithmic scales for the cases in which $\alpha(t) = t^{-\frac{1}{2}}, t^{-\frac{1}{4}}, t^{-\frac{1}{8}}$ in Figure 2. As can be seen, $\mathbf{x}(t)$ converges to $\mathbf{1}_4 \otimes \mathbf{y}^*$, which verifies the correctness of Theorem 1. Further, according to the trajectories in Figure 2, we calculate the slopes $\kappa = -0.5035, \kappa = -0.2497, \kappa = -0.1242$ for these three cases, respectively. Also we plot the trajectory of $e(t)$ for $\alpha(t) = t^{-1}$ and obtain the slope $\kappa = -0.4667$.

These results suggest that when $\alpha(t)$ is taken as $\alpha(t) = t^{\kappa}$ for $-1 < \kappa < 0$, the trajectories of the solution $\mathbf{x}(t)$ to (4) will tend to the least squares solution to (1) with the error $e(t)$ decreasing at an rate $\mathcal{O}(t^{\kappa})$. Surprisingly, $\alpha(t) = t^{-1}$ seems to be an exception, with which the error $e(t)$ is decreasing by $\mathcal{O}(t^{-0.4667})$.

Example 2. Consider the same linear equation as in Example 1, where $\mathbf{x}(0) = [0.3\ -0.2\ 0.1\ -0.1\ 1.1\ 1.6\ -0.2\ 0.8]^\top$ and $\alpha(t) = t^{-\frac{1}{2}}$. Let (4) do iteration over the graph in Figure 1 for $K = 1, 10, 100$, respectively. Then we plot the trajectories of $e(t) = \|\mathbf{x}(t) - \mathbf{1}_4 \otimes \mathbf{y}^*\|$ in logarithmic scales in Figure 3. In all three cases the solution $\mathbf{x}(t)$ of (4) converges to $\mathbf{1}_4 \otimes \mathbf{y}^*$, consistent with Theorem 1. Also we calculate the slopes κ for the cases $K = 1, 10, 100$ and obtain $\kappa = -0.4924, \kappa = -0.5024, \kappa = -0.5035$.

These results illustrate that the asymptotic convergence speed of the flow (4) is dominated by the term $\alpha(t)$, while the consensus gain K plays a role in affecting the transient behaviors of the trajectories.

B. Switching Connected Graphs

Example 3. Consider the following linear equation with respect to $\mathbf{y} \in \mathbb{R}^2$:

$$\begin{bmatrix} 3 & 2 \\ 1 & -3 \\ 1 & 5 \\ -1 & 4 \\ 2 & 4 \end{bmatrix} \mathbf{y} = \begin{bmatrix} 1 \\ 1 \\ 5 \\ 3 \\ -2 \end{bmatrix}.$$

As can be calculated, its unique least squares solution is $\mathbf{y}^* = [-0.2008\ 0.4344]^\top$. Let $\mathcal{Q}^* = \{\mathcal{G}_1, \mathcal{G}_2\}$ with $\mathcal{G}_1, \mathcal{G}_2$ as shown in Figure 4 and $\mathcal{G}_{\sigma(t)}$ be given as following:

$$\mathcal{G}_{\sigma(t)} = \begin{cases} \mathcal{G}_1,\ t \in \big[Tk, T(k+1)\big), k = 0, 2, 4, \dots \\ \mathcal{G}_2,\ t \in \big[Tk, T(k+1)\big), k = 1, 3, 5, \dots \end{cases}$$

with $T = 1$, i.e., the network switches between graph $\mathcal{G}_1$ and $\mathcal{G}_2$ periodically with period $T = 1$. Set the initial value $\mathbf{x}(0) = [-1\ 1\ -0.5\ 1\ -1.2\ 0.6\ -0.3\ -0.4\ -1\ 0.3]^\top$. Let the flow (4) do iteration over the switching network $\mathcal{G}_{\sigma(t)}$ with $K = 100, \alpha(t) = (t+1)^{-1}$. It can be known by simple calculation that the conditions of Theorem 2 are met, in particular, $\mathrm{rank}(\mathbf{H}) = 2$. Then we plot the trajectories $\mathbf{x}_i[1](t), \mathbf{x}_i[2](t)$ with $i = 1, 2, 3, 4, 5$ in Figure 5. We can see that $\mathbf{x}_i(t)$ for all i converge to $\mathbf{y}^*$, consistent with Theorem 2.

Example 4. Consider the following linear equation with respect to $\mathbf{y} \in \mathbb{R}^2$:

$$\begin{bmatrix} 4 & 6 \\ 1 & 1.5 \\ 1 & 1.5 \\ -1 & -1.5 \\ 2 & 3 \end{bmatrix} \mathbf{y} = \begin{bmatrix} 2 \\ -1 \\ 1 \\ -3 \\ 2.5 \end{bmatrix}.$$

Let the network $\mathcal{G}_{\sigma(t)}, K, \alpha(t)$ be the same as in Example 3. We can easily know that the conditions of Theorem 2 are satisfied, in particular, $\mathrm{rank}(\mathbf{H}) = 1 < 2$, which means the linear equation has non-unique least squares solutions. Set the initial value $\mathbf{x}(0) = [-1\ 1\ -0.5\ 1\ -1.2\ 0.6\ -0.3\ -0.4\ -1\ 0.3]^\top$. Let (4) do iteration under these conditions. Then the trajectories of $\mathbf{x}_i[1](t), \mathbf{x}_i[2](t)$ with $i = 1, 2, 3, 4, 5$ are plotted in Figure 6, from which it can seen that $\mathbf{x}_i(t)$ for all i converge to $\hat{\mathbf{y}} = [-0.5705\ 0.8442]^\top$. Evidently, $\hat{\mathbf{y}}$ is one of the least squares solutions and this validates the result in Theorem 2.

C. Switching Graphs with Joint Connectivity

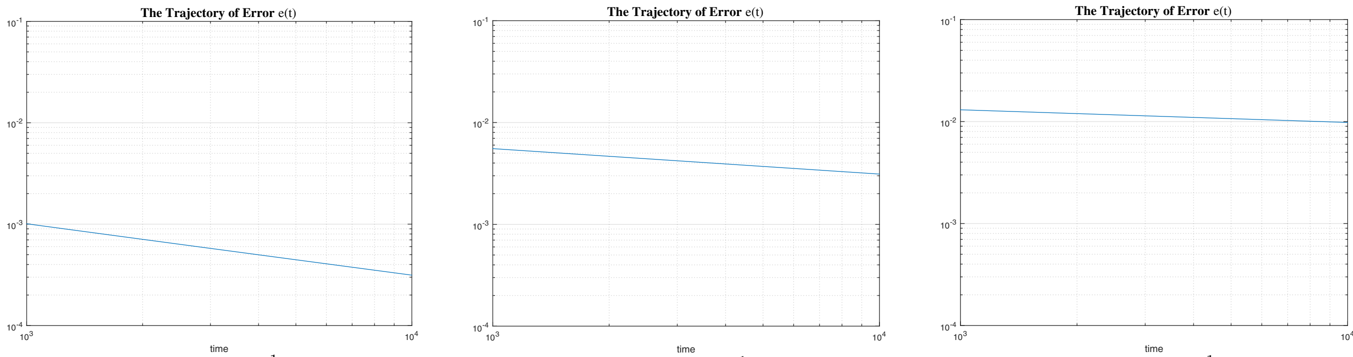
Example 5. Consider the same linear equation as Example 3. Let $\mathcal{G}_{\sigma(t)}$ be given as following:

$$\mathcal{G}_{\sigma(t)} = \begin{cases} \mathcal{G}_3,\ t \in \big[Tk, T(k+1)\big), k = 0, 2, 4, \dots \\ \mathcal{G}_4,\ t \in \big[Tk, T(k+1)\big), k = 1, 3, 5, \dots \end{cases}$$

with $\mathcal{G}_3$, $\mathcal{G}_4$ in Figure 7, $T = 1$. We can see that neither $\mathcal{G}_3$ nor $\mathcal{G}_4$ is connected, but only $\mathcal{G}_3 \cup \mathcal{G}_4$ is connected. Given the same $K, \alpha(t), \mathbf{x}(0)$ as Example 3. Let the flow (4) do iteration over $\mathcal{G}_{\sigma(t)}$. Then we plot the trajectories of $\mathbf{x}_i[1](t), \mathbf{x}_i[2](t)$ for all i in Figure 8. It can be seen that $\mathbf{x}_i(t)$ converge to $\mathbf{y}^* = [-0.2008\ 0.4344]^\top$ for all i when $\mathrm{rank}(\mathbf{H}) = m$. We can also verify the convergence for the case with $\mathrm{rank}(\mathbf{H}) < m$.

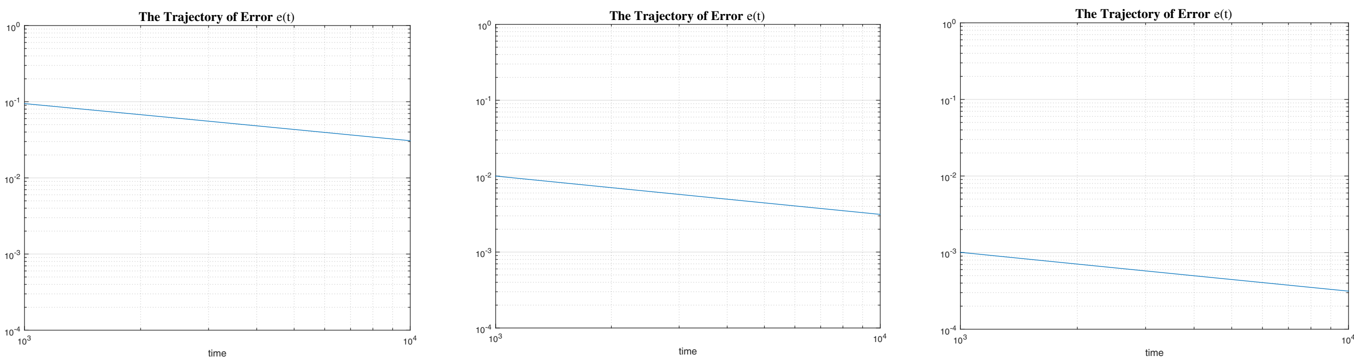
V. CONCLUSIONS AND FUTURE WORKS

In this paper, a first-order distributed step-size least squares solver over networks was proposed. When the least squares solution is unique, We proved the convergence results for fixed and connected graphs, with an assumption of nonintegrable step size. By loosening the uniqueness of the least squares solution, we obtained the convergence results for constantly connected switching graphs, with square integrable step size. We also provided a few numerical examples, in order to verify the results and illustrate the convergence speed. Potential future work includes proving the convergence over networks without instantaneous connectivity, studying the exact convergence rate, and finding out the convergence limit.



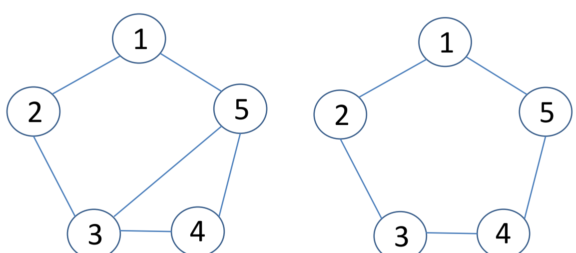
(a) $\alpha(t) = t^{-\frac{1}{2}}$ (b) $\alpha(t) = t^{-\frac{1}{4}}$ (c) $\alpha(t) = t^{-\frac{1}{8}}$

Fig. 2. The trajectories of $e(t) = \|\mathbf{x}(t) - \mathbf{1}_4 \otimes \mathbf{y}^*\|$ in logarithmic scale. The slope $\kappa = -0.5035, -0.2497, -0.1242$ for $\alpha(t) = t^{-\frac{1}{2}}$, $\alpha(t) = t^{-\frac{1}{4}}$, $\alpha(t) = t^{-\frac{1}{8}}$, respectively.



(a) K=1 (b) K=10 (c) K=100

Fig. 3. The trajectories of $e(t) = \mathbf{x}(t) - \mathbf{1}_4 \otimes \mathbf{y}^*$ in logarithmic scale. The slopes $\kappa = -0.4924, -0.5024, -0.5035$ for $K = 1, 10, 100$, respectively.



(a) $\mathcal{G}_1$ (b) $\mathcal{G}_2$

Fig. 4. Constant, connected and undirected graph $\mathcal{G}_1$, $\mathcal{G}_2$ considered in Example 3 and 4.

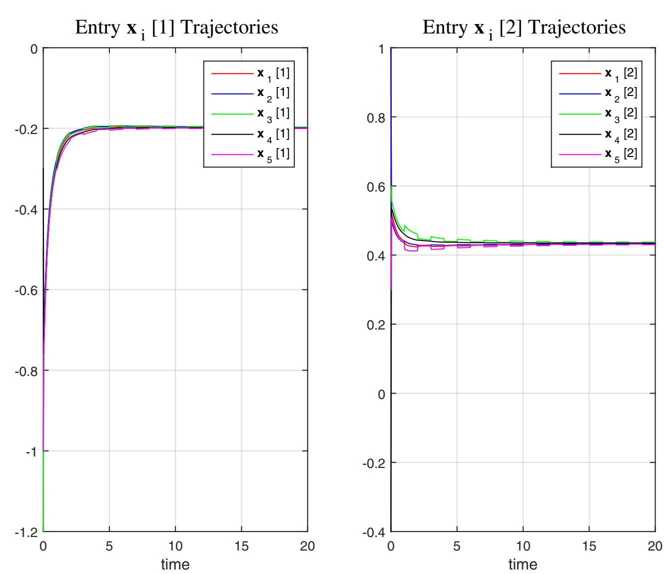


Fig. 5. The trajectories of $\mathbf{x}_i[1](t), \mathbf{x}_i[2](t)$ for $i = 1, 2, 3, 4, 5$ given $K = 100$, $\alpha(t) = (t+1)^{-1}$ obtained over a switching network. The result shows all $\mathbf{x}_i(t)$ converge to $\mathbf{y}^* = [-0.2008\ 0.4344]^\top$.

References

[1] R. Olfati-Saber and R. M. Murray, "Consensus problems in networks of agents with switching topology and time-delays," *IEEE Transactions on automatic control*, vol. 49, no. 9, pp. 1520–1533, 2004.

[2] A. Nedic and A. Ozdaglar, "Distributed subgradient methods for multi-agent optimization," *IEEE Transactions on Automatic Control*, vol. 54, no. 1, pp. 48–61, 2009.

[3] F. S. Cattivelli, C. G. Lopes, and A. H. Sayed, "Diffusion recursive least-squares for distributed estimation over adaptive networks," *IEEE Transactions on Signal Processing*, vol. 56, no. 5, pp. 1865–1877, 2008.

[4] P. Barooah and J. P. Hespanha, "Graph effective resistance and distributed control: Spectral properties and applications," in *Decision and control, 2006 45th IEEE conference on*, pp. 3479–3485, IEEE, 2006.

[5] S. Kar and J. M. Moura, "Gossip and distributed kalman filtering:

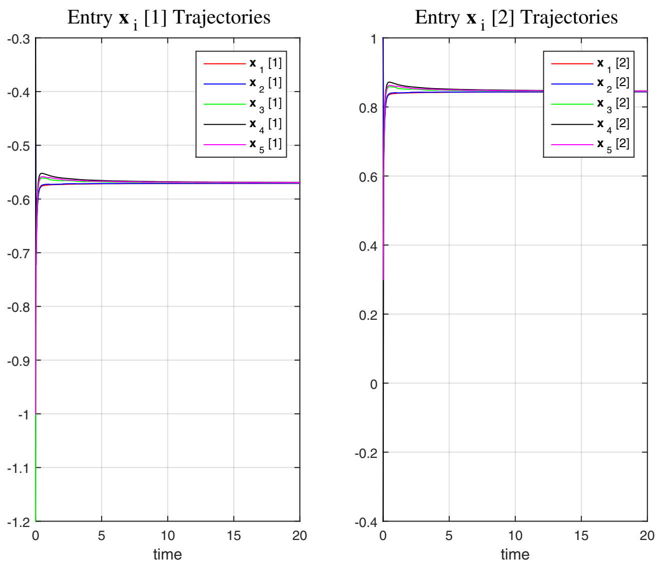
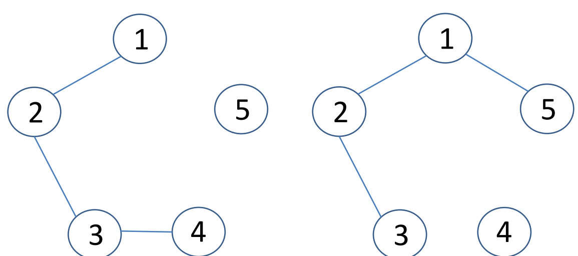


Fig. 6. The trajectories of the first component $\mathbf{x}_i[1](t)$ and the second component $\mathbf{x}_i[2](t)$ for $i = 1,2,3,4,5$ given $K = 100$, $\alpha(t) = (t+1)^{-1}$ obtained over a switching network. As calculated, all $\mathbf{x}_i(t)$ converge to $\hat{\mathbf{y}} = [-0.5705\ 0.8442]^\top$, which is one of the least squares solutions.



(a) $\mathcal{G}_3$ (b) $\mathcal{G}_4$

Fig. 7. Constant, connected and undirected graph $\mathcal{G}_3$, $\mathcal{G}_4$ considered in Example 5.

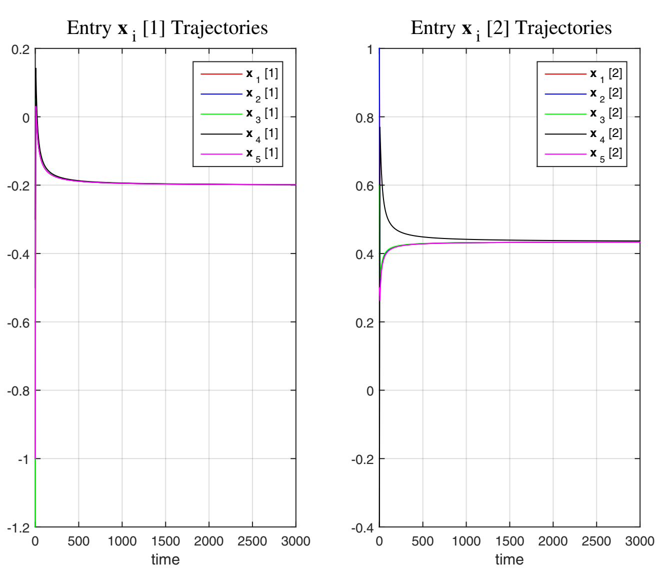


Fig. 8. The trajectories of the first component $\mathbf{x}_i[1](t)$ and the second component $\mathbf{x}_i[2](t)$ for $i = 1,2,3,4,5$ given $K = 100$, $\alpha(t) = (t+1)^{-1}$ obtained over a switching network with connected graph union. It can be seen that all $\mathbf{x}_i(t)$ converge to $\mathbf{y}^* = [-0.2008\ 0.4344]^\top$.

Weak consensus under weak detectability," *IEEE Transactions on Signal Processing*, vol. 59, no. 4, pp. 1766–1784, 2011.
[6] C. Wang, K. Ren, J. Wang, and Q. Wang, "Harnessing the cloud for securely outsourcing large-scale systems of linear equations," *IEEE Transactions on Parallel and Distributed Systems*, vol. 24, no. 6, pp. 1172–1181, 2013.
[7] B. Anderson, S. Mou, A. S. Morse, and U. Helmke, "Decentralized gradient algorithm for solution of a linear equation," *arXiv preprint arXiv:1509.04538*, 2015.
[8] G. Shi, B. D. Anderson, and U. Helmke, "Network flows that solve linear equations," *IEEE Transactions on Automatic Control*, vol. 62, no. 6, pp. 2659–2674, 2017.
[9] J. Lu and C. Y. Tang, "Distributed asynchronous algorithms for solving positive definite linear equations over networksPart i: Agent networks," *IFAC Proceedings Volumes*, vol. 42, no. 20, pp. 252–257, 2009.
[10] J. Liu, S. Mou, and A. S. Morse, "An asynchronous distributed algorithm for solving a linear algebraic equation," in *52nd IEEE Conference on Decision and Control*, pp. 5409–5414, 2013.
[11] S. Mou and A. Morse, "A fixed-neighbor, distributed algorithm for solving a linear algebraic equation," in *Proc. European Control Conference*, pp. 2269–2273, 2013.
[12] S. Mou, J. Liu, and A. S. Morse, "A distributed algorithm for solving a linear algebraic equation," *IEEE Transactions on Automatic Control*, vol. 60, no. 11, pp. 2863–2878, 2015.
[13] J. Wang and N. Elia, "Solving systems of linear equations by distributed convex optimization in the presence of stochastic uncertainty," *IFAC Proceedings Volumes*, vol. 47, no. 3, pp. 1210–1215, 2014.
[14] J. Wang and N. Elia, "Distributed solution of linear equations over unreliable networks," in *American Control Conference (ACC), 2016*, pp. 6471–6476, IEEE, 2016.
[15] J. Wang and N. Elia, "Control approach to distributed optimization," in *Communication, Control, and Computing (Allerton), 2010 48th Annual Allerton Conference on*, pp. 557–561, Sept 2010.
[16] J. Wang and N. Elia, "Distributed least square with intermittent communications," in *American Control Conference (ACC), 2012*, pp. 6479–6484, IEEE, 2012.
[17] B. Gharesifard and J. Cortés, "Distributed continuous-time convex optimization on weight-balanced digraphs," *IEEE Transactions on Automatic Control*, vol. 59, no. 3, pp. 781–786, 2014.
[18] Y. Liu, C. Lageman, B. Anderson, and G. Shi, "Exponential least squares solvers for linear equations over networks," *World Congress of the International Federation of Automatic Control*, 2017.
[19] A. Nedić and A. Olshevsky, "Distributed optimization over time-varying directed graphs," *IEEE Transactions on Automatic Control*, vol. 60, no. 3, pp. 601–615, 2015.
[20] A. Nedic, A. Ozdaglar, and P. A. Parrilo, "Constrained consensus and optimization in multi-agent networks," *IEEE Transactions on Automatic Control*, vol. 55, no. 4, pp. 922–938, 2010.
[21] R. A. Horn and C. R. Johnson, *Matrix analysis*. Cambridge university press, 2012.