Distributed Cooperative Learning Over Networks via Fuzzy Logic Systems

Pengfei Ren[1], Weisheng Chen[1,2]
1. School of Mathematics and Statistics, Xidian University, Xian, 710126
E-mail: rocketpengfei@yeah.net
2. School of Aerospace Science and Technology, Xidian University, Xian, 710126
E-mail: wshchen@126.com

Abstract: In this paper, we expand the scope of research on distributed cooperative learning (DCL) via fuzzy logic systems (FLSs) over an undirected and connected network, that is, each node (or learner) cooperatively learn an unknown pattern (or function) and finally reach consensus through local information interaction with their one-hop neighbors. Based on the approximation of FLSs, we present continuous-time and discrete-time DCL algorithms with respect to the aforementioned problem. The optimal coefficient matrix of the FLS is trained by the two proposed algorithms, respectively. And we use algebraic graph theory and Lyapunov approach to prove that the weight coefficient matrix in the algorithms has an exponential convergence rate, respectively. Moreover, the algorithms reduce communication cost and bandwidth compared with the centralized learning (CL) algorithms. Some simulation results illustrate the effectiveness and advantages of the proposed algorithms.

Key Words: Distributed cooperative learning (DCL), consensus, fuzzy logic systems (FLSs), continuous-time, discrete-time, Lyapunov approach, exponential convergence.

1 INTRODUCTION

Distributed in-network data processing based on the one-hop communication of networks, such as wireless sensor networks (WSNs), has attracted significant attention in recent years. In this paper, we focus on the problem of distributed cooperative learning (DCL) via a fuzzy logic system (FLS) over a network, which is undirected and connected. Under the framework of supervised machine learning, we consider a network of spatially distributed sensors over a geographic area, in which all nodes (or learners) cooperatively learn an unknown pattern (or function) through the approximation of an FLS. Meanwhile, we train the optimal coefficient matrix of the FLS with the proposed DCL algorithms in order to minimize the global approximation error function (the cost function) and make all the nodes reach consensus.

We are motivated to do the research by the approximation capability and performance of an FLS [3]–[7] and, more importantly, the development of distributed learning [8]–[17]. Previous researches have proposed various distributed algorithms, most of which are under the framework of adaptive processing and cooperative strategies, including the incremental type (e.g., incremental LMS [8], [9], incremental RLS [11], [12]), the diffusion type (e.g., diffusion LMS [13], [14], diffusion RLS [17]), and the average consensus (the space diffusion) type [15], [16], etc. Although the incremental algorithms minimize the communication cost, they have their own disadvantages, that is, in the incremental methods, a cyclic path is needed over the nodes in the network to make the estimates broadcasted from node to node, and it's an NP-hard problem to determine such a Hamilton circle traversing all the nodes in the network. The diffusion algorithms only use local information interaction between single-hop neighbors and their learning rate are faster than the incremental ones. Whereas in many diffusion algorithms, each node in the network minimizes the local cost function by using the gradient descent method, which makes the estimate converge too slowly near the minimizer value. Moreover, the smaller constant step-sizes make the better learning performance and also make the diffusion algorithms converge slowly. The average consensus algorithms use the gradually decreasing step-sizes to make all nodes reach the same minimizer value but prevent them from learning when the step-sizes are reduced to 0. And many consensus algorithms need to define the ”bridge nodes” in the network. In our proposed algorithms, the general consensus theory [10] is incorporated. Similar to the diffusion algorithms, each non-hierarchical node only transmits its locally learnt knowledge to its one-hop neighbors. The initial value of each node is the local optimal value and the learning process keeps the sum of the gradient of each local cost function at 0. By using the algebraic graph theory [1], [2] and the Lyapunov approach, the weight coefficient matrix of the FLS is proved to converge exponentially to the optimal value, which is faster than the diffusion algorithms. The DCL algorithms equal to the conventional centralized learning (CL) ones in performance.

The remainder of this paper is organized as follows. Section II presents preliminaries, including notation, algebraic

This work is supported by the National Natural Science Foundation of China (61174213, 61503292).

graph theory and approximation via FLSs. In section III, we present the continuous-time and discrete-time DCL algotithms. In Section IV, we give some simulations to verify the effectiveness and advantages of the DCL algorithms. At last, some conclusions are summarized in Section V.

2 Preliminaries

2.1 Notation

In this study, $\mathbb{R}$ denotes the set of real numbers; $\mathbb{R}^n$ denotes the set of $n \times 1$ real vectors; $\mathbb{R}^{n\times n}$ denotes the set of $n \times n$ real matrices; $\mathbb{N}$ denotes the set of all natural numbers; I^n denotes the $n \times n$ identity matrix; $\mathbf{1}_n$ is an n-dimensional vector with all ones; A^T denotes the transpose of the matrix A; $\otimes$ denotes the Kronecker product; $\|\cdot\|$ denotes the Euclidean norm; ∇f is the gradient of *f*, and $\nabla^2 f$ is the Hessian matrix of *f*.

2.2 Algebraic Graph Theory

In this paper, we model a communication network by introducing an undirected graph of N nodes, $\mathcal{G} \triangleq \{\mathcal{V}, \mathcal{E}, \mathcal{A}\}$, where $\mathcal{V} = \{\upsilon_1, \upsilon_2, \cdots, \upsilon_N\}$ is a finite nonempty node set, $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$ is the edge set and $\mathcal{A} = [a_{ij}]_{N\times N} \in \mathbb{R}^{N\times N}$ is the weighted adjacency matrix of $\mathcal{G}$ with $a_{ij} \geq 0$ and $a_{ij} = a_{ji}$. For simplicity, $i \in \mathcal{V}$ means node $\upsilon_i \in \mathcal{V}$. An edge in $\mathcal{G}$ is expressed as $(\upsilon_j, \upsilon_i) \in \mathcal{E}$, which means $(\upsilon_i, \upsilon_j) \in \mathcal{E}$ for arbitrary $i, j \in \mathcal{V}$. If $(\upsilon_i, \upsilon_j) \in \mathcal{E}$, there is information interaction between node *i* and node *j*, and node *j* is called a neighbor of node *i*, furthermore, $a_{ij} > 0$, otherwise, $a_{ij} = 0$. Assume that there is no loop at each node, i.e., $a_{ii} = 0$. The Laplacian matrix of $\mathcal{G}$ is defined as $\mathcal{L} \triangleq \mathcal{D} - \mathcal{A}$, in which $\mathcal{D} = \text{diag}(d_1, d_2, \cdots, d_N)$ and $d_i = \sum_{j=1}^{N} a_{ij}$. Node i's neighbor set is defined as $\mathcal{N}_i = \{j \in \mathcal{V} \mid (\upsilon_i, \upsilon_j) \in \mathcal{E}\}$, in which $|\mathcal{N}_i|$ represents the cardinality of $\mathcal{N}_i$. Note that $\mathcal{L}$ is real symmetric, so $\mathcal{L} = \mathcal{K} \times \mathcal{K}^T$, where $\mathcal{K}$ is the incidence matrix of an arbitrary orientation of $\mathcal{L}$ [2]. Further, $\mathcal{L}$ is positive semidefinite and hence all eigenvalues of $\mathcal{L}$ are nonnegative. Meanwhile, $\mathcal{L}$ has zero column sums, therefore $\mathcal{L} \times \mathbf{1}_N = 0$ and further $\lambda_1(\mathcal{L}) = 0$. The multiplicity of zero as an eigenvalue of $\mathcal{L}$ is the number of the connected components of $\mathcal{G}$, and so if $\mathcal{G}$ is connected, $\lambda_2(\mathcal{L})$ is the smallest nonzero eigenvalue, that is, $0 = \lambda_1(\mathcal{L}) < \lambda_2(\mathcal{L}) \leq \cdots \leq \lambda_N(\mathcal{L})$. Just for the Laplacian matrix $\mathcal{L}$, we rewrite $\lambda_i(\mathcal{L})$ as λ_i for simplicity, $i = 1, 2, ..., N$.

2.3 Approximation via Fuzzy Logic Systems

An multi-input single-output (MISO) FLS maps from the input space $U \subset \mathbb{R}^m$ to the outspace $V \subset \mathbb{R}$. Suppose, $U = U_1 \times U_2 \times \cdots \times U_m$, in which $U_p \subset \mathbb{R}, p = 1, 2, \cdots, m$. And the diagram of an FLS is shown in Fig. 1.

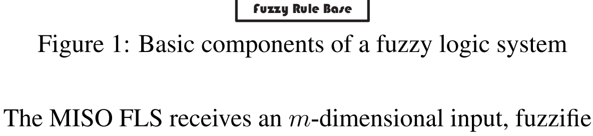


Figure 1: Basic components of a fuzzy logic system

The MISO FLS receives an m-dimensional input, fuzzifies it, maps the gotten fuzzy sets in U to a fuzzy set in V via the fuzzy inference engine, performs the defuzzification, and derives the output. Both the input and output are nonfuzzy. The fuzzy rule base is composed of fuzzy IF-THEN rules as follows:

$\mathbf{R_j}$: **IF** x_1 is M_1^j, **and** x_2 is M_2^j, **and** ... **and** x_m is M_m^j,
THEN y is N^j.

in which $x = [x_1, x_2, \cdots, x_m] \in U$ is the m-dimensional input, along with $x_p \in U_p$ and $y \in V$ is the output of the FLS. $M_p^j \subset U_p$ and $N^j \subset U$ are fuzzy sets, which are respectively characterized by the membership functions $\mu_{M_p^j}(x_p)$ and $\mu_{N^j}(y)$.

For any real continuous function $f(x)$, $f(x)$ on a compact set can be approximated to arbitrary accuracy by an FLS:

$$f(x) = s(x)W_i + \varepsilon_i \tag{1}$$

where $\hat{f}_i(x) = s(x)W_i$ is the ith estimation function of the objective function $f(x)$, W_i, the weight coefficient to be trained, denotes the approximation of the optimal weight coefficient W^*. As W_i is a function of time t or k, here we use W_i for short. $s(x)$ is the chosen fuzzy basis function (FBF), which composes the basis vector function $S(x)$, and ε_i is an infinitesimal. Then, as an example, we rewrite (1) into a specific form as follows:

$$\begin{aligned} f(x) &= [s_1(x), s_2(x), \cdots, s_n(x)] \begin{bmatrix} w_{i1} \\ w_{i2} \\ w_{i3} \\ \vdots \\ w_{in} \end{bmatrix} + \sum_{j=1}^{n} \varepsilon_{ij} \\ &= s(x)W_i + \varepsilon_i \end{aligned} \tag{2}$$

where $a_{pj} \in (0, 1]$, $\sigma_{pj} \in (0, +\infty)$ and x_{pj} are turnable parameters. $s(x) = [s_1(x), s_2(x), \cdots, s_n(x)] \in \mathbb{R}^{1\times n}$ along with $s_j(x) = \frac{\prod_{p=1}^{m} a_{pj} exp\left(-(\frac{x_p - x_{pj}}{\sigma_{pj}})^2\right)}{\sum_{j=1}^{n}\prod_{p=1}^{m} a_{pj} exp\left(-(\frac{x_p - x_{pj}}{\sigma_{pj}})^2\right)}$, and $\varepsilon_i = \sum_{j=1}^{n} \varepsilon_{ij}$.

3 Distributed Cooperative Learning via Fuzzy Logic Systems

3.1 Problem Statement

For each node $i \in \mathcal{V}$, we assume that, N_i, the size of the labeled training set $Z_i = \left\{(x_i^l, y_i^l)\right\}_{l=1}^{N_i}$, is available. $\mathcal{N}_i$ is node *i*'s neighbor set. The sum of training samples throughout the network is $T = \sum_{i=1}^{N} N_i$. Our purpose is to find a technique to identify the common objective function $f(x)$ optimally, that is, $f_i(x) = f(x), \forall i \in \mathcal{V}$, by using T training samples and our contributions is to approximate $f(x)$ using the DCL algorithms via FLSs. For node i, the form of approximation through the FLSs is shown as follows:

$$f(x) = \sum_{j=1}^{n} w_{ij} s_j(x) + \varepsilon_i = s(x)W_i + \varepsilon_i \tag{3}$$

The estimation error function of node i is defined as follows:

$$F_i(W_i) = \sum_{k=1}^{N_i} \|y_i^k - s(x)W_i\|^2 = \|Y_i - S_i(x)W_i\|^2 \tag{4}$$

in which $Y_i = [y_i^1, y_i^2, \cdots, y_i^{N_i}]_{N_i\times 1}^T$ and $S_i(x) = \mathbf{1}_{N_i} s(x) \in \mathbb{R}_{N_i\times n}$.

Hence, the global approximation error function is:

$$F(W) = \sum_{i=1}^{N} F_i(W_i) = \sum_{i=1}^{N} \|Y_i - S_i W_i\|^2 \qquad (5)$$

Obviously, $F(W)$ shows the total approximation degree between the sample outputs and the values of the approximation functions. For all nodes $i \in \mathcal{V}$, the smaller the $F(W)$ is, the closer to $f(x)$ is the estimation function $\hat{f}_i(x)$.

Motivated by the distributed learning based on the consensus theory and the approximation capability and performance of FLSs, we present consensus-based DCL algorithms to train the optimal coefficient matrix W^* of the FLS via local information exchange in order to achieve the optimal approximation. Before using the DCL algorithms, we design the following method to minimize both the norm of weight coefficients W_i and the estimation error:

$$\begin{aligned} \min_W G(W) &= \frac{1}{2}\sum_{i=1}^{N}(\|Y_i - S_i W_i\|^2 + \sigma_i \|W_i\|^2) \\ &= \sum_{i=1}^{N} g_i(W_i) \end{aligned} \qquad (6)$$

in which $g_i(W_i) = \frac{1}{2}(\|Y_i - S_i W_i\|^2 + \sigma_i \|W_i\|^2)$, $\sigma_i > 0$ is an adjustable parameter and it can be regarded as a trade-off between the norm of weight coefficients W_i and the estimation error. Obviously, (6) is a quadric convex optimization problem, so there must exist a unique value W^*, which makes $G(W)$ reach the bottom and satisfies $\sum_{i=1}^{N} \nabla g_i(W^*) = 0$.

3.2 Continuous-Time DCL Algorithm

Our ultimate purpose is shown as follows:

$$\lim_{t \to +\infty} W_i(t) = W_i^*, \forall i \in \mathcal{V}. \qquad (7)$$

Assume that the undirected graph $\mathcal{G}$ is connected, then the continuous-time DCL algorithm for training weight coefficient matrix is expressed as follows:

$$\begin{cases} \dot{W}_i(t) = \gamma({S_i}^T S_i + \sigma_i I_n)^{-1}\Big[\sum\limits_{j \in \mathcal{N}_i} a_{ij}\big(W_j(t) - W_i(t)\big)\Big] \\ W_i(0) = ({S_i}^T S_i + \sigma_i I_n)^{-1} S_i^T Y_i, \quad \forall t \geq 0, \forall \in \mathcal{V}. \end{cases} \qquad (8)$$

where $W_i(0)$ is the initial value of $W_i(t)$ and $\gamma > 0$ is a tunable parameter.

To perform the convergence analysis of the algorithm (8) in ease, for all nodes in $\mathcal{G}$, the algorithm (8) can be rewritten as:

$$\begin{cases} \dot{W}(t) = -\gamma(S^T S + \sigma \otimes I_n)^{-1}(\mathcal{L} \otimes I_n) W(t) \\ W(0) = (S^T S + \sigma \otimes I_n)^{-1} S^T Y, \forall t \geq 0, \forall i \in \mathcal{V}. \end{cases} \qquad (9)$$

where $W(t) = [W_1^T(t), W_2^T(t), \cdots, W_N^T(t)]^T \in \mathbb{R}^{Nn \times 1}$, $S = diag\{S_1, S_2, \cdots, S_N\} \in \mathbb{R}^{T \times Nn}$, $\sigma = diag\{\sigma_1, \sigma_2, \cdots \sigma_N\} \in \mathbb{R}^{N \times N}$, and $Y = [Y_1^T, Y_2^T, \cdots, Y_N^T]^T \in \mathbb{R}^{T \times 1}$.

Theorem 1: Consider the continuous-time DCL algorithm (8) under the connectivity hypothesis, then we can prove that the following inequality satisfies:

$$\sum_{i=1}^{N} \|W^* - W_i(t)\|^2 \leq \frac{2}{\xi} V\big(W(0)\big) e^{-\frac{2\lambda_2 \gamma}{\Xi} t} \qquad (10)$$

where $\xi_i = \lambda_{min}(S_i^T S_i + \sigma_i I_n)$ along with $\xi = \min\limits_{i \in \mathcal{V}} \xi_i$; and $\Xi_i = \lambda_{max}(S_i^T S_i + \sigma_i I_n)$ along with $\Xi = \max\limits_{i \in \mathcal{V}} \Xi_i$.

So, our purpose is reached that $\lim\limits_{t \to \infty} W_i(t) = W^*, \forall i \in \mathcal{V}$ by using the continuous-time DCL algorithm. And hence, $\hat{f}_i(x) = S_i(x) W^*$ optimally approximates $f(x)$.

Proof: See Appendix A. ■

3.3 Discrete-Time DCL Algorithm

In this subsection, we propose the discrete-time DCL algorithm. Similar to the continuous-time form, our goal is:

$$\lim_{t \to \infty} W_i(k) = W^*, \forall i \in \mathcal{V}. \qquad (11)$$

Provided that the undirected graph $\mathcal{G}$ is connected, then the discrete-time DCL algorithm is expressed as follows:

$$\begin{cases} W_i(k+1) = \gamma({S_i}^T S_i + \sigma_i I_n)^{-1}\Big[\sum_{j \in \mathcal{N}_i} a_{ij}\big(W_j(k) \\ \qquad\qquad - W_i(k)\big)\Big] + W_i(k) \\ W_i(0) = ({S_i}^T S_i + \sigma_i I_n)^{-1} S_i^T Y_i, \forall k \in \mathbb{N}, \forall i \in \mathcal{V} \end{cases} \qquad (12)$$

where $k \in \mathbb{N}$ is a discrete point in time. Meanwhile, we also rewrite the algorithm (12) in the following matrix form:

$$\begin{cases} W(k+1) = -\gamma(S^T S + \sigma \otimes I_n)^{-1}(\mathcal{L} \otimes I_n) W(k) + W(k) \\ W(0) = (S^T S + \sigma \otimes I_n)^{-1} S^T Y, \ \forall t \geq 0, \forall i \in \mathcal{V} \end{cases} \qquad (13)$$

in which $W(k) = [W_1^T(k), W_2^T(k), \cdots, W_N^T(k)]^T \in \mathbb{R}^{Nn \times 1}$.

Theorem 2: Consider the discrete-time DCL algorithm (12) under the connectivity hypothesis, then we can prove that the following inequality satisfies:

$$\sum_{i=1}^{N} \|W^* - W_i(k)\|^2 \leq \frac{2}{\xi} V\big(W(0)\big) \theta^k \qquad (14)$$

where $\theta = 1 - \frac{2\lambda_2 \gamma}{\Xi}\big(1 - \frac{\lambda_N \gamma}{\xi}\big)$.

Thus, our purpose is reached that $\lim\limits_{t \to +\infty} W_i(k) = W^*, \forall i \in \mathcal{V}$ by using the discrete-time DCL algorithm. And hence, $\hat{f}_i(x) = S_i(x) W^*$ optimally approximates $f(x)$.

Proof: See Appendix B. ■

4 Simulation experiments

The objective function approximation using the DCL Algorithms via fuzzy logic systems

In this subsection, we use simulation experiments to verify the performance of the proposed two DCL algorithms, respectively. And the objective function is specified as follows:

$$f(x) = \frac{x}{1 + x^2}, x \in [-5, 5] \qquad (15)$$

Consider an undirected connected network in Fig. 2, where the Laplacian matrix is: $\mathcal{L}$=[4,-1,-1,-1,-1;-1,3,-1,0,-1;-1,-1,3,-1,0;-1,0,-1,3,-1;-1,-1,0,-1,3]. Each node's own training set Z_i, which are randomly assigned and uniformly distributed in the interval [-5,5]. And we set that every node gets $N_i = 10000$ training samples. The training set of each node and the objective function are illustrated in Fig. 3.

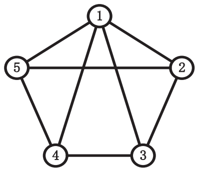


Figure 2: The topological graph

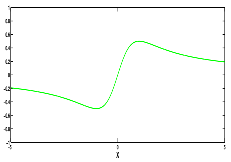
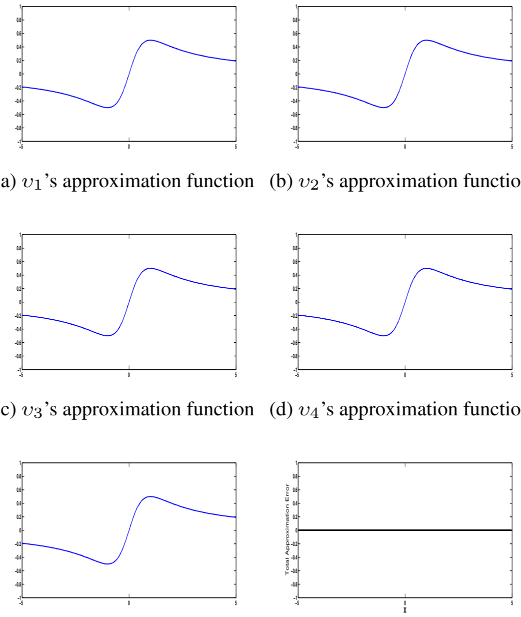


Figure 3: The objective function

We randomly divide both the input and output space into n=100 fuzzy sets, which are respectively represented as $A_1, A_2, ..., A_{100}$ and $B_1, B_2, ..., B_{100}$, generate n=100 fuzzy rules from the sample data as the fuzzy rule base, and assign every fuzzy set a corresponding FBF in both the input and output space. The initial center values of the output sets in the fuzzy rules at each node are different.

In both cases of the continuous-time and discrete-time DCL algorithms, we set γ=0.003 and σ_i=0.01 for the objective function approximation. Results of the simulation experiments in the two cases are illustrated in Fig. 4 and Fig. 5, respectively.



(a) υ_1's approximation function (b) υ_2's approximation function

(c) υ_3's approximation function (d) υ_4's approximation function

(e) υ_5's approximation function (f) The total approximation error

Figure 4: Results of simulation with the continuous-time DCL algorithm via fuzzy logic systems: (a)-(e): the approximation function of υ_1-υ_5, (f): the total approximation error.

The vector of the output sets' center values in the fuzzy rules at each node, that is, W_i, finally reaches consensus, and the ultimate fuzzy rules are:

$\mathbf{R_1}$: **IF** x is A_1: $[-5, -4.9480]$, **THEN** y is B_1, whose center value is -0.1144;

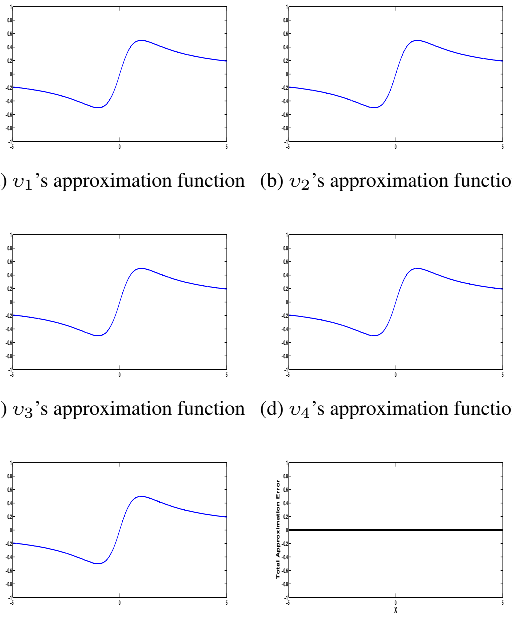
$\mathbf{R_2}$: **IF** x is A_2: $(-4.9480, -4.7650]$, **THEN** y is B_2, whose

center value is -0.2099;

......

$\mathbf{R_{100}}$: **IF** x is A_{100}: $[4.7550, 5]$, **THEN** y is B_{100}, whose center value is 0.0087.

Since FLSs are universal approximators, whatever the objective function is, no matter the continuous-time or discrete-time DCL algorithm, if we set reasonable fuzzy rules and appropriate inner parameters, the approximation performance is always precise.



(a) υ_1's approximation function (b) υ_2's approximation function

(c) υ_3's approximation function (d) υ_4's approximation function

(e) υ_5's approximation function (f) The total approximation error

Figure 5: Results of simulation with the discrete-time DCL algorithm via fuzzy logic systems: (a)-(e): the approximation function of υ_1-υ_5, (f): the total approximation error.

5 Conclusion

The main contribution of this article is that we apply the strategy of consensus-based DCL to pattern recognition via FLSs. The problem we study, built on the quadric convex optimization, is the problem of supervised machine learning, i.e., all the spatially distributed nodes learn the unknown pattern over the network and optimize the global error function. The proposed DCL algorithms are used to train the optimal efficient matrix, the weight, through merely local information exchange and all the learners reach consensus in the end. The information for interaction is the learnt knowledge of each node, but not the raw partial data. We prove that the weight exponentially converges to the optimal value in the DCL algorithms. In contrast to the CL algorithm, communication cost and width are reduced by the DCL algorithms. There are still many interesting and challenging problems deserving further research. And in the context of big data, the presented DCL algorithms are realistic in many applications, such as wireless sensor networks (WSNs), including environment monitoring, traffic control and so on.

6 appendices

6.1 Proof of the Theorem 1

Proof: In order to perform the convergence analysis of the continuous-time DCL algorithm, we first construct a Lyapunov function candidate as follows:

$$V\big(W(t)\big)=\frac{1}{2}\sum_{i=1}^{N}\big(W^*-W_i(t)\big)^T\big(S_i^TS_i+\sigma_iI_n\big)\big(W^*-W_i(t)\big) \quad (16)$$

And we can further derive:

$$V\big(W(t)\big)\geq\sum_{i=1}^{N}\frac{\xi_i}{2}\|W^*-W_i(t)\|^2\geq\frac{\xi}{2}\sum_{i=1}^{N}\|W^*-W_i(t)\|^2 \quad (17)$$

where $\xi_i = \lambda_{min}(S_i^TS_i + \sigma_iI_n)$ and $\xi = \min_{i\in\mathcal{V}} \xi_i$.

Meanwhile, we can also get the following inequality (Proof see Appendix 6.3):

$$V\big(W(t)\big) \leq \frac{\Xi}{2\lambda_2} W(t)^T(\mathcal{L}\otimes I_n)W(t) \tag{18}$$

Consider the Lyapunov function candidate $V\big(W(t)\big)$, we can obtain the derivative of it is:

$$\begin{aligned}\frac{dV\big(W(t)\big)}{dt} &= -\sum_{i=1}^{N} \dot{W}_i(t)^T(S_i^T S_i + \sigma_i I_n)\big(W^* - W_i(t)\big)\\ &= -\gamma W(t)^T(\mathcal{L}\otimes I_n)W(t)\end{aligned} \tag{19}$$

Along with (18), (19) turns into

$$\frac{dV\big(W(t)\big)}{dt} \leq -\frac{2\lambda_2\gamma}{\Xi} V\big(W(t)\big) \tag{20}$$

Then, it is easy to derive:

$$V\big(W(t)\big) \leq V\big(W(0)\big)e^{-\frac{2\lambda_2\gamma}{\Xi}t} \tag{21}$$

Combining (21) with (17), yields:

$$\sum_{i=1}^{N}\|W^* - W_i(t)\|^2 \leq \frac{2}{\xi}V\big(W(t)\big) \leq \frac{2}{\xi}V\big(W(0)\big)e^{-\frac{2\lambda_2\gamma}{\Xi}t} \tag{22}$$

Finally, $W_i(t)$ converges exponentially to W^*. ■

Remark 1: From the algorithm (8), under the connectivity hypothesis, it is effortless to obtain that $\sum_{i=1}^{N}(S_i^T S_i + \sigma_i I_n)\dot{W}_i(t) = \gamma \sum_{i=1}^{N}\sum_{j\in\mathcal{N}_i} a_{ij}\big(W_j(t) - W_i(t)\big) = 0$.

6.2 Proof of the Theorem 2

Proof: Consider the form of (16), similarly, we can easily establish the Lyapunov function candidate as follows, which is used to perform the convergence analysis of the discrete-time DCL algorithm:

$$V\big(W(k)\big) = \frac{1}{2}\sum_{i=1}^{N}\big(W^* - W_i(k)\big)^T(S_i^T S_i + \sigma_i I_n)\big(W^* - W_i(k)\big) \tag{23}$$

And further, we can derive:

$$V\big(W(k)\big) \geq \sum_{i=1}^{N}\frac{\xi_i}{2}\|W^* - W_i(k)\|^2 \geq \frac{\xi}{2}\sum_{i=1}^{N}\|W^* - W_i(k)\|^2 \tag{24}$$

In the meanwhile, we can also obtain the following inequality (Proof process refer to Appendix 6.3):

$$V\big(W(k)\big) \leq \frac{\Xi}{2\lambda_2} W(k)^T(\mathcal{L}\otimes I_n)W(k) \tag{25}$$

Consider the Lyapunov function candidate, we can easily derive the difference of it is:

$$\begin{aligned}&\triangle V\big(W(k+1)\big) = V\big(W(k+1)\big) - V\big(W(k)\big)\\ =& -\frac{1}{2}\sum_{i=1}^{N}\big(W_i(k)^T(S_i^T S_i + \sigma_i I_n)W_i(k)\\ &- W_i(k+1)^T(S_i^T S_i + \sigma_i I_n)W_i(k+1)\big)\end{aligned} \tag{26}$$

Remark 2: From the algorithm (12), under the connectivity hypothesis, it is easy to obtain that $\sum_{i=1}^{N}(S_i^T S_i + \sigma_i I_n)\big(W_i(k+1) - W_i(k)\big) = \gamma \sum_{i=1}^{N}\sum_{j\in\mathcal{N}_i} a_{ij}\big(W_j(k) - W_i(k)\big)$=0.

For ease of simplification, we construct intermediate terms in the function, so it leads to:

$$\begin{aligned}&\triangle V\big(W(k+1)\big)\\ \leq& \sum_{i=1}^{N}\big(W_i(k+1) - W_i(k)\big)^T(S_i^T S_i + \sigma_i I_n)W_i(k+1)\\ =& -\gamma W(k)^T(\mathcal{L}\otimes I_n)W(k+1)\end{aligned} \tag{27}$$

United with (13), (27) turns into

$$\begin{aligned}&\triangle V\big(W(k+1)\big) = V\big(W(k+1)\big) - V\big(W(k)\big)\\ \leq& -\gamma W(k)^T(\mathcal{L}\otimes I_n)[-\gamma(S^T S + \sigma\otimes I_n)^{-1}\\ &\times(\mathcal{L}\otimes I_n)W(k) + W(k)]\\ \leq& -\gamma\big(1 - \frac{\lambda_N\gamma}{\xi}\big)W(k)^T(\mathcal{L}\otimes I_n)W(k)\end{aligned} \tag{28}$$

On the condition that γ satisfies $1 - \frac{\lambda_N\gamma}{\xi} > 0$, that is, $0 < \gamma < \frac{\xi}{\lambda_N}$, plus $V\big(W(k)\big)$ is non-negative, which is proved before, we can further deduce that $\{V\big(W(k)\big)\}_{i=1}^{+\infty}$ is a non-increasing sequence. Combining (28) with (25), yields:

$$V\big(W(k+1)\big) - V\big(W(k)\big) \leq -\frac{2\lambda_2\gamma}{\Xi}\big(1 - \frac{\lambda_N\gamma}{\xi}\big)V\big(W(k)\big) \tag{29}$$

Next, it is effortless to obtain that:

$$V\big(W(k+1)\big) \leq \theta V\big(W(k)\big) \tag{30}$$

where $\theta = 1 - \frac{2\lambda_2\gamma}{\Xi}\big(1 - \frac{\lambda_N\gamma}{\xi}\big)$.

In order to let $\{V\big(W(k)\big)\}_{i=1}^{+\infty}$ be a non-increasing sequence, we set $0 < \gamma < \frac{\xi}{\lambda_N}$ to make $0 < 1 - \frac{\lambda_N\gamma}{\xi} < 1$. Moreover, to make (29) continue to maintain this feature, we set $0 < \theta < 1$ and it is easy to work out that $0 < \gamma < \frac{\Xi}{2\lambda_2}$. Hence, together with the calculation results before, we set $0 < \gamma < \min\{\frac{\xi}{\lambda_N}, \frac{\Xi}{2\lambda_2}\}$, which guarantees that $\{V\big(W(k)\big)\}_{i=1}^{+\infty}$ is both non-negative and non-increasing. Based on the recurrence relations (30), we employ the inverse extrapolation method to get the relation between $V\big(W(k)\big)$ and $V\big(W(0)\big)$ as follows:

$$V\big(W(k)\big) \leq \theta V\big(W(k-1)\big) \leq \cdots \leq \theta^k V\big(W(0)\big) \tag{31}$$

Combining (31) with (24), results in

$$\sum_{i=1}^{N}\|W^* - W_i(k)\|^2 \leq \frac{2}{\xi}V\big(W(k)\big) \leq \frac{2}{\xi}\theta^k V\big(W(0)\big) \tag{32}$$

In the end, $W_i(k)$ converges exponentially to W^*. ■

6.3 Proof of the Inequality (18)

Lemma 1: For the undirected and connected network topology $\mathcal{G}$, if $\sum_{i=1}^{N}\nabla^2 g_i\big(W_i(t)\big) = 0$ and $V\big(W(t)\big) = \sum_{i=1}^{N}\Big[g_i(W^*) - g_i\big(W_i(t)\big) - \nabla g_i\big(W_i(t)\big)^T\big(W^* - W_i(t)\big)\Big] = 0$, the following inequality is satisfied:

$$\begin{aligned}V(W(t)) \;\leq\;& \sum_{i=1}^{N}\frac{\Xi_i}{2}\|W_i(t) - \frac{1}{N}\sum_{j=1}^{N}W_j(t)\|^2\\ \leq\;& \frac{\Xi}{2}\sum_{i=1}^{N}\|W_i(t) - \frac{1}{N}\sum_{j=1}^{N}W_j(t)\|^2\end{aligned} \tag{33}$$

where $\Xi_i = \lambda_{max}(\nabla g_i^2)$ and $\Xi = \max_{i\in\mathcal{V}}\Xi_i$.

Lemma 2: For the undirected and connected graph set $\mathbb{G}$, $\mathcal{G}\in\mathbb{G}$ with N nodes and $\bar{\mathcal{G}}$ is the complete graph of $\mathcal{G}$. Similar to $\mathcal{L}$, we define $\bar{\mathcal{L}}\in\mathbb{R}^{N\times N}$ is the Laplacian matrix of $\bar{\mathcal{G}}$. It's easy to derive that $\bar{\mathcal{L}}$ has $N-1$ positive eigenvalues at N, and $\mathcal{L}$ has $N-1$ positive eigenvalues among which λ_2 is the smallest, and there exists a $W \in \mathbb{R}^{N\times N}$, which contains N orthonormal eigenvalues of $\mathcal{L}$ in its columns. Then, $\lambda_2 W^T\bar{\mathcal{L}}W \le NW^T\mathcal{L}W$, which makes $\lambda_2\bar{\mathcal{L}} \le N\mathcal{L}$.

Proof: From remark 1, we get $\sum_{i=1}^{N}(S_i^TS_i + \sigma_i I_n)\dot{W}_i(t) = \sum_{i=1}^{N}\nabla^2 g_i\big(W_i(t)\big)=0$. So $\sum_{i=1}^{N}\nabla g_i\big(W_i(t)\big)$ is constant. Further, we can obtain $\sum_{i=1}^{N}\nabla g_i\big(W_i(t)\big) = \sum_{i=1}^{N}\nabla g_i\big(W_i(0)\big)=\sum_{i=1}^{N}[-S_i^TY_i+S_i^TS_i+\sigma_i I_n W_i(0)]= \sum_{i=1}^{N}-S_i^TY_i+(S_i^TS_i+\sigma_i I_n)(S_i^TS_i+\sigma_i I_n)^{-1}S_i^TY_i=0$ for any $t\in\mathcal{V}$. Therefore, we can derive the optimal coefficient value W^* of $G(W)$.

Then, according to the quadric convex problem (6), we get $g_i(W_i) = \frac{1}{2}(\|Y_i - S_iW_i\|^2 + \sigma_i\|W_i\|^2)$, further we derive:

$$\begin{aligned}&\sum_{i=1}^{N}\Big[g_i(W^*)-g_i\big(W_i(t)\big)-\nabla g_i\big(W_i(t)\big)^T\big(W^*-W_i(t)\big)\Big]\\ &=\sum_{i=1}^{N}\Big[\frac{1}{2}\big(W^*-W_i(t)\big)^T(S_i^TS_i+\sigma_i I_n)\big(W^*-W_i(t)\big)\Big]\\ &=V\big(W(t)\big) \end{aligned}\tag{34}$$

Next, in the light of Lemma 2, it is easy to derive:

$$\begin{aligned}&\sum_{i\in\mathcal{V}}\|W_i(t)-\frac{1}{N}\sum_{i\in\mathcal{V}}W_j(t)\|^2\\ &=\frac{1}{N}[W_1(t)^T,W_2(t)^T,\cdots,W_N(t)^T](\bar{\mathcal{L}}\otimes I_n)\\ &\quad\times[W_1(t)^T,W_2(t)^T,\cdots,W_N(t)^T]^T\\ &=\frac{1}{N}W(t)^T(\bar{\mathcal{L}}\otimes I_n)W(t)\\ &\le\frac{1}{\lambda_2}W(t)^T(\mathcal{L}\otimes I_n)W(t)\end{aligned}\tag{35}$$

And hence, combine with Lemma 1, we have:

$$V(W(t)) \le \frac{\Xi}{2\lambda_2}W(t)^T(\mathcal{L}\otimes I_n)W(t) \tag{36}$$

Remark 3[1]: For the graph set $\mathbb{G}$, the function $\lambda_2(\mathcal{L}_G)$ is non-decreasing for arbitrary graph $G\in\mathbb{G}$ with the same set of vertices, i.e. $\lambda_2(\mathcal{L}_{G_1}) \le \lambda_2(\mathcal{L}_{G_2})$ if $G_1\subseteq G_2$ (G_1 and G_2 have the same set of vertices). Obviously, $\mathcal{G}\subseteq\bar{\mathcal{G}}$. And hence, $\lambda_2(\mathcal{L}_{\mathcal{G}})\le\lambda_2(\mathcal{L}_{\bar{\mathcal{G}}})$.

Remark 4[1]: For the graph $\bar{\mathcal{G}}$ with N nodes, $\lambda_2(\mathcal{L}_{\bar{\mathcal{G}}})=N$, while for the non-complete graph $\mathcal{G}$ with N nodes, $\lambda_2(\mathcal{L}_{\mathcal{G}})\le N$. Therefore, $\lambda_2(\mathcal{L}_{\bar{\mathcal{G}}}) = \lambda_3(\mathcal{L}_{\bar{\mathcal{G}}}) = \cdots = \lambda_N(\mathcal{L}_{\bar{\mathcal{G}}}) = N$.

REFERENCES

[1] M. Fiedler, Algebraic connectivity of graphs. Czechoslovak mathematical journal, Vol.23, No.2, 298-305, 1973.

[2] C. G, G. R, Algebraic graph theory, Graduate Texts in Mathematics, vol.207, 2001.

[3] L. X. Wang, Stable adaptive fuzzy control of nonlinear systems, IEEE Trans. on Fuzzy Systems, Vol.1, No.2, 146-155, 1993.

[4] L. X. Wang, Fuzzy systems are universal approximators, IEEE International Conference on Fuzzy Systems, 1163-1170, 1992.

[5] L. X. Wang, J. M. Mendel, Fuzzy basis functions, universal approximation, and orthogonal least-squares learning, IEEE Trans. on Neural Networks, Vol.3, N0.5, 807-814, 1992.

[6] X. J. Zeng, M. G. Singh, Approximation theory of fuzzy systems-SISO case, IEEE Trans. on Fuzzy Systems, Vol.2, No.2, 162-176, 1994.

[7] L. X. Wang, J. M. Mendel, Generating fuzzy rules by learning from examples, IEEE Trans. on. Systems, Man and Cybernetics, Vol.22, No.6, 1414-1427, 1992.

[8] F. S. Cattivelli, A. H. Sayed, Analysis of spatial and incremental LMS processing for distributed estimation, IEEE Transactions on Signal Processing, Vol.59 ,No.4, 1465-1480, 2011.

[9] A. Khalili, M. A. Tinati, A. Rastegarnia, Steady-state analysis of incremental LMS adaptive networks with noisy links, IEEE Transactions on Signal Processing, Vol.59, No.5, 2416-2421, 2011.

[10] W. Chen, S. Hua, S. S. Ge, Consensus-based distributed cooperative learning control for a group of discrete-time nonlinear multi-agent systems using neural networks, Automatica, Vol.50, No.9, 2254-2268, 2014.

[11] A. Khalili, M. A. Tinati, A.Rastegarnia, Analysis of incremental RLS adaptive networks with noisy links, IEICE Electronics Express, Vol.8, No.9, 623-628, 2011.

[12] A. H. Sayed, C. G. Lopes, Distributed recursive least-squares strategies over adaptive networks, IEEE Fortieth Asilomar Conference on Signals, Systems and Computers, 233-237, 2006.

[13] F. S. Cattivelli, A. H. Sayed, Diffusion LMS strategies for distributed estimation, IEEE Trans. on Signal Processing, Vol.58, No. 3, 1035-1048, 2010.

[14] C. G. Lopes, A. H. Sayed, Diffusion least-mean squares over adaptive networks: Formulation and performance analysis, IEEE Trans. on Signal Processing, Vol.56, No.7, 3122-3136, 2008.

[15] L. Xiao, S. Boyd, S. Lall, A space-time diffusion scheme for peer-to-peer least-squares estimation, Proceedings of the 5th international conference on Information processing in sensor networks, 168-176, 2006.

[16] L. Georgopoulos, M. Hasler, Distributed machine learning in networks by consensus, Neurocomputing, Vol.124, 2-12, 2014.

[17] F. S. Cattivelli, C. G. Lopes, A. H. Sayed, A diffusion RLS scheme for distributed estimation over adaptive networks, IEEE 8th Workshop on Signal Processing Advances in Wireless Communications, 1-5, 2007.