23rd International Symposium on Mathematical Theory of Networks and Systems
Hong Kong University of Science and Technology, Hong Kong, July 16-20, 2018

Distributed Optimization for Multi-agent Systems of Independent Step Size*

Dong Wang and Yangwei Chen

***Abstract*— In this paper, we develop a distributed algorithm for multiple agents aiming to solve an optimization problem. The communication among the agents is governed by an undirected graph. In our algorithm, each agent uses an independent step-size and the upper bound of steps are independent from network topologies. The step-sizes depend on local objective functions. We present an algorithm that even with different step sizes guarantees convergence of the agent states to a common point. With such step range, the states of each agent converge to a common optimal point at a linear rate $O(\kappa^{-k})$ where $0<\kappa<1$ and k is the number of iterations. As compared to existing approaches, we only need the local cost function of each agent is convex rather than strongly convex. Finally, the numerical examples are given to validate the theoretical results obtained.**

***Index Terms*— Distributed optimization; Multi-agent systems; Independent step-size.**

I. Problem Formulation and Algorithm

A. Proposed algorithm

Notation. The notation used here is standard except where otherwise stated. We use a capital letter to define a matrix and a lowercase bold letter to define a vector. The $n\times n$ unit matrix is represented by I. The n-dimensional vectors of all of the elements are 1 is represented by $\mathbf{1}_n$. We use $\mathbb{R}^n$ and $\mathbb{R}^{n\times m}$ to define the n-dimensional space and the $n\times m$ real matrices, respectively. For any vector $\mathbf{x}$, $\mathbf{x}_i$ is its ith element. Similarly, A_i is the ith row, and A_{ij} is the (i,j)th component of any a matrix A. For a given matrix G and vector $\mathbf{x}$, we denote the G-matrix norm $\|\mathbf{x}\|_G^2=\langle \mathbf{x}, G\mathbf{x}\rangle$. If G is positive definite, we have $\|\mathbf{x}\|_G^2>0$. In order to simplify the representation, we use $A\geq 0$ to indicate that A is Positive Semi-Definite (PSD) and $A>0$ to explain that matrix A is Positive Definite (PD). The maximum and minimum eigenvalues of matrix A are defined as $\lambda_{\max}(A)$ and $\lambda_{\min}(A)$. The minimum non-zero eigenvalue of matrix A is defined by $\tilde{\lambda}_{\min}(A)$. $diag\{x_1,x_2,...,x_n\}$ is the diagonal matrix with the ith diagonal entry x_i. $\|\cdot\|$ denotes the Euclidean norm. The gradient vector of a function f is denoted as ∇f.

B. Problem formulation

Consider an undirected graph defined as $\mathscr{G}(\mathscr{V},\mathscr{E})$, where $\mathscr{V}=\{1,2,3,...,n\}$ is a set of n nodes, representing agents and $\mathscr{E}\subseteq\mathscr{V}\times\mathscr{V}$ is a set of edges composed with ordered pair (i,j), $i,j\in\mathscr{V}$, where agent i and agent j exchange information with each other. Define an adjacency matrix $A=[a_{ij}]$ of graph $\mathscr{G}$ with $a_{ij}>0$ if $(i,j)\in\mathscr{E}$ and $a_{ij}=0$ otherwise.

In this paper, we consider distributed optimization problems of multi-agent systems, where the cost function is a summation of local cost functions of all agents. Suppose that agent i is assigned with the states variable $x_i\in\mathbb{R}^m$, and each agent has a local cost function f_i, $\forall i\in\mathscr{V}$. Agent i needs to exchange information with its neighbors such that the states of all agent ultimately converge to a common optimum point. The optimization problem model is represented as

$$\min\ f(\mathbf{x})=\sum_{i=1}^{n} f_i(\mathbf{x}), \tag{1}$$

where the function f_i: $\mathbb{R}^d\to\mathbb{R}$ is convex differentiable with the Lipschitz continuous gradients and known only by agent i. Suppose that the optimal solution set of Problem (1) is nonempty.

In this section, a distributed gradient optimization algorithm with an independent step size is proposed for multi-agent systems to solve problem (1). Before proposing the algorithm, let's start with some assumptions and properties which are used for our results.

Assumption 1: The graph $\mathscr{G}$ is an undirected and connected graph. The adjacency matrix $A=[a_{ij}]\in\mathbb{R}^{n\times n}$ satisfies $A=A^T$, $\mathbf{1}_n^T A=1$ and $A\geq 0$.

One way to make sure that $A\geq 0$ is true is to design the mixing matrix A to be strictly diagonally dominant. Following Assumption 1, we define a stochastic matrix $\bar{A}=\frac{1}{3}I+\frac{2}{3}A$. It follows from Assumption 1 that $\bar{A}\geq\frac{I}{3}$.

Assumption 2: The local cost function f_i of agent i is closed convex, and the gradient ∇f_i, $i\in\mathscr{V}$ is Lipschitz continuous with a positive constant L_i and satisfies

$$\|\nabla f_i(\mathbf{x}_i)-\nabla f_i(\mathbf{y}_i)\|\leq L_i\|\mathbf{x}_i-\mathbf{y}_i\|,\forall\mathbf{x}_i,\mathbf{y}_i\in\mathbb{R}^d.$$

Following Assumption 2, we define the gradient collection $\nabla f(\mathbf{x})\triangleq[\nabla f_1(\mathbf{x}_1);\ldots;\nabla f_n(\mathbf{x}_n)]^T$ for $\mathbf{x}\triangleq[\mathbf{x}_1;\ldots;\mathbf{x}_n]^T$. For $\forall\mathbf{x},\mathbf{y}\in\mathbb{R}^{nd}$, it follows from the convex function $f(\mathbf{x})$ that,

$$\|\nabla f(\mathbf{x})-\nabla f(\mathbf{y})\|_{L^{-1}}^2\leq\langle\mathbf{x}-\mathbf{y},\nabla f(\mathbf{x})-\nabla f(\mathbf{y})\rangle,$$

where $L=diag\{L_1,\cdots,L_n\}$.

We are in the position to propose the following distributed

*This work was supported in part by the National Natural Science Foundation of China under Grant 61473055, Grant 61374070 and Grant 61773089, by the Liaoning Province Science Foundation under Grant 2015020075, by Fundamental Research Funds for the Central Universities under Grant DUT17ZD227, and by Youth Star of Dalian Science and Technology under Grant 2015R052 and Grant 2016RQ014.

Dong Wang and Yangwei Chen are Control Science and Engineering, Dalian University of Technology, Dalian, P. R. China dwang@dlut.edu.cn

algorithm to solve problem (1) with independent step size:

$$\mathbf{x}_i^{k+1} = \sum_{j=1}^{n} \bar{a}_{ij}\mathbf{x}_i^k - \alpha_i \mathbf{y}_i^k,$$
$$\mathbf{y}_i^{k+1} = \nabla f_i(\mathbf{x}_i^{k+1}) - \beta \sum_{t=0}^{k+1}\sum_{j=1}^{n} (a_{ij} - \bar{a}_{ij})\mathbf{x}_j^t. \quad (2)$$

In order to better simplify the expression, we write (2) in a compact form. Let the weights a_{ij} and $\bar{a}_{ij}$ be the elements of the matrices A and $\bar{A}$, i.e., $A = [a_{ij}]$ and $\bar{A} = [\bar{a}_{ij}]$. Let $\mathbf{x}^k$, $\nabla f(\mathbf{x}^k)$ be a collection of states and gradients of all agents in time k, i.e., $\mathbf{x}^k \triangleq [\mathbf{x}_1^k; \mathbf{x}_2^k; \ldots; \mathbf{x}_n^k]^T$, $\nabla f(\mathbf{x}^k) \triangleq [\nabla f_1(\mathbf{x}_1^k); \nabla f_2(\mathbf{x}_2^k); \ldots; \nabla f_n(\mathbf{x}_n^k)]^T$. Then the algorithm (2) is rewritten by

$$\mathbf{x}^{k+1} = \bar{A}\mathbf{x}^k - D\mathbf{y}^k,$$
$$\mathbf{y}^{k+1} = \nabla f(\mathbf{x}^{k+1}) - \beta \sum_{t=0}^{k+1} (A - \bar{A})\mathbf{x}^t. \quad (3)$$

Where A and $\bar{A}$ are doubly stochastic matrices and $D = diag\{\alpha_1, \cdots, \alpha_n\}$. α_i is the step size of agent i.

Remark 1: During the iteration, agent i receives the weighting states $a_{ij}\mathbf{x}_j^{k+1}$, and $\bar{a}_{ij}\mathbf{x}_j^k$ from its neighbors. Each agent choose their own step size α_i, and α_i is allowed to be different. In the proposed algorithm, the constant β is chosen such that $(2I_n - \beta D)(I_n - A) \geq 0$. In order to get an exact optimal solution, we use difference values of gradient. The second equation of algorithm (3) is an accurate gradient estimate. It modifies the impact of the independent step size with the cumulative state information.

II. Main results

In this section, we present the linear convergence result of the proposed algorithm (3). With the appropriate step size, the algorithm converges to the optimal solution of the problem (1).

For simplicity, we define the auxiliary sequence $\mathbf{q}^k = \sum_{r=0}^{k} \mathbf{x}^r$ and $\mathbf{q}^* = \lim_{k\to\infty} \mathbf{q}^k$.

Lemma 1: () Suppose that Assumptions 1-2 hold, $\mathbf{x}^*$ is solution of the problem (1) if the following properties are satisfied,

$$(A - \bar{A})\mathbf{x}^* = 0,$$
$$\mathbf{1}^T \nabla f(\mathbf{x}^*) = 0. \quad (4)$$

In Lemma 1, we give the optimal condition of the proposed algorithm, that is, the state of the optimal solution. Next, we give the relationship between the state variables and the optimal solution.

Lemma 2: Suppose that Assumptions 1-2 hold. The quadruple sequence $\{\mathbf{x}^k, \mathbf{x}^*, \mathbf{q}^k, \mathbf{q}^*\}$ satisfies the following property,

$$M(\mathbf{x}^{k+1} - \mathbf{x}^*) + D^{-1}\bar{A}(\mathbf{x}^{k+1} - \mathbf{x}^k)$$
$$= \beta(A - \bar{A})(\mathbf{q}^{k+1} - \mathbf{q}^*) - (\nabla f(\mathbf{x}^k) - \nabla f(\mathbf{x}^*)), \quad (5)$$

where $M = D^{-1}(I - \bar{A} + \beta D(A - \bar{A}))$.

In fact,M is also written as

$$M = D^{-1}(I - \bar{A} + \beta D(A - \bar{A}))$$
$$= D^{-1}(\frac{2(I - A)}{3} + \frac{\beta D}{3}(A - I))$$
$$= \frac{1}{3}D^{-1}(2I - \beta D)(I - A), \quad (6)$$

then matrix M is positive definite by the definition of constant β in remark (1).

Next, we state the important results in this paper. We first establish a relationship between $\mathbf{q}^k$, $\mathbf{q}^*$, $\mathbf{x}^k$ and $\mathbf{x}^*$.

Based on $\mathbf{q}^k$, $\mathbf{q}^*$, $\mathbf{x}^k$ and $\mathbf{x}^*$, we define the following notations to simplify calculation:

$$\mathbf{u}^k = \begin{bmatrix} \mathbf{x}^k \\ \mathbf{q}^k \end{bmatrix}, \mathbf{u}^* = \begin{bmatrix} \mathbf{x}^* \\ \mathbf{q}^* \end{bmatrix}, G = \begin{bmatrix} N & 0 \\ 0 & \beta U \end{bmatrix},$$

where $N = D^{-1}\bar{A}$ and $U = \bar{A} - A$.

Theorem 1: Suppose that Assumptions 1-2 hold and each agent updates their states according to (3), then the sequence $\{\mathbf{u}^k\}$ satisfies

$$0 \leq \left\|\mathbf{u}^k - \mathbf{u}^*\right\|_G^2 - \left\|\mathbf{u}^{k+1} - \mathbf{u}^*\right\|_G^2 \quad (7)$$

and the step size α_i satisfies

$$\alpha_i < 2/3L_i. \quad (8)$$

Remark 2: It is noted that the positive definite matrix is important to analyze the convergence of the designed algorithm in [1]. By the property of a positive definite matrix, we define a non-negative matrix norm. The method is also used in our proof. In Theorem 1, the sequence $\{\mathbf{u}^k\}$ converges to $\mathbf{u}^*$ with the G-matrix norm. The algorithm (3) is used for an independent step size. When proving the convergence of the proposed algorithm, we need that the gradient of cost function is Lipschitz continuous and the strong convex assumption is not needed.

Next, The convergence rate result is given in the theorem.

Theorem 2: Suppose that Assumptions 1-2 hold. The iterative sequences $\mathbf{u}^k$ satisfies

$$\left\|\mathbf{u}^k - \mathbf{u}^*\right\|_G^2 \geq (1 + \eta)\left\|\mathbf{u}^{k+1} - \mathbf{u}^*\right\|_G^2, \quad (9)$$

where

$$\eta = \min\left\{\frac{c_1}{c_3 c_4}, \frac{c_2}{c_3 c_5}\right\}, \quad (10)$$

and

$$c_1 = \frac{\beta\tilde{\lambda}_{\min}(U)\tilde{\lambda}_{\min}(U^2)}{\lambda_{\max}(U^2)} + 2\tilde{\lambda}_{\min}(M),$$
$$c_2 = \lambda_{\min}(D^{-1}\bar{A} - L/2),$$
$$c_3 = \frac{\lambda_{\max}(U)}{\beta\tilde{\lambda}_{\min}(U^2)},$$
$$c_4 = 4\lambda_{\max}(M^2) + 4L_{\max} + \lambda_{\max}(P),$$
$$c_5 = 4\lambda_{\max}(PP^T) + 4L_{\max}.$$

That is, $\{\mathbf{u}^k\}$ converges to $\{\mathbf{u}^*\}$ at R-linear rate.

Remark 3: It is worth mentioning that there are many algorithms solving the distributed optimization problem with the diminishing step size [2], [3], [4], [5]. The rates of convergence are usually slower than their similar algorithms in the centralized optimization. The diminishing step $\alpha_k = k^{-1/2}$ gives rise to the convergence rate $O(\frac{\ln k}{\sqrt{k}})$ [2]. The distributed nesterov method has a faster convergence rate $O(\frac{\ln k}{k})$ with the diminishing step size $\alpha_k = k^{-1/3}$ [4]. There are strict restrictions of the diminishing step size, which needs the condition of non-summation and square-summation. In order to accelerate the convergence rate, a constant step size is used. It does not only need to use the global topology information, but also limit the range of the step size. Comparing with these algorithms, we relax the limit of selection range of step sizes and consider the condition of the independent step size. We present the linear convergence rate of our algorithm in Theorem 2.

III. Numerical Example

In this section, we consider the decentralized sensing problems to study the convergence rate of our algorithm with least-squares problem in a directed graph. Each agent has its own cost function $(C_i x - c_i)^2$, which is only known by itself. In the simulation, we use five sensors to cooperatively estimate a parameter $\mathbf{x}$. We use nodes instead of sensors and consider distributed optimization problem with a network of 5 agents. The agents exchange informations with a undirected graph $\mathscr{G}$ in the following example described in Figure 1. This optimization problem is modeled as follows,

$$\mathbf{x}^* \leftarrow \arg\min_{x\in\mathbb{R}} f(\mathbf{x}) = \sum_{i=1}^{n} f_i(x),$$

where $f_i(x) = \|C_i x - c_i\|_2^2, C_i \in R^{n_i \times p}, c_i \in R^{n_i}, for\ i = 1,2,\ldots,n.$

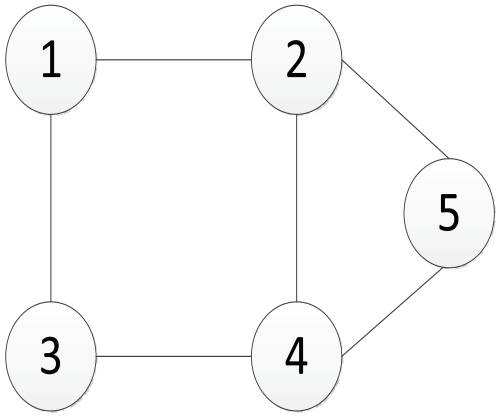

Fig. 1. An directed and connected graph for the agents in the numerical example.

In order to illustrate that our proposed algorithm has a rapid convergence speed. Similarly, we consider a group of five agents, exchanging information among the agents is described via an undirected and connected graph as Figure 1 shows. Utilizing the same mixing matrices, the proposed algorithm (3) is compared with DGD [6] and EXTRA [1]. The simulation result is shown in Figure 2. Let DGD use the diminishing step size. At the beginning of execution, the convergence rate of DGD is close to linear. But as time goes on, the step size decreases and the rate of convergence slows down. When the same constant step is used, the proposed algorithm (3) is better performed comparing with the EXTRA. Our algorithm can use the independent step size. The rate of convergence is accelerated by changing the step size of each agent.

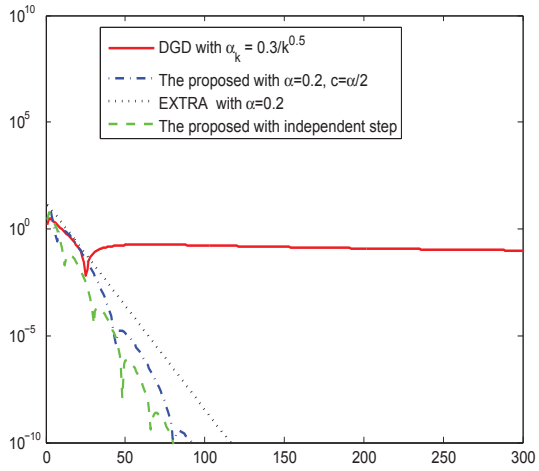

Fig. 2. Comparison of DGD and the proposed algorithm (3). Our algorithm has a faster convergence rate.

IV. Conclusion

In this paper, a distributed optimization algorithm is proposed to solve optimization problem of a multi-agent system with independent step size. We show that the proposed algorithm (3) for the distributed optimization problem (1) converges geometrically to to a common optimal point when each agent uses a different constant step size. Moreover, we also give an accurate convergence rate related to the step size. When the convergence of the algorithm is given, we only use the Lipschitz continuous of the cost function without the assumption of strong convexity. Finally, the theoretical results have ultimately been testified by mean of a numerical simulation example. Through a comparison, it has been demonstrated that the algorithm we have established is faster convergent than the DGD and EXTRA.

References

[1] W. Shi, Q. Ling, G. Wu, and W. Yin, "Extra: An exact first-order algorithm for decentralized consensus optimization," *SIAM Journal on Optimization*, vol. 25, no. 2, pp. 944–966, 2015.
[2] I.-A. Chen *et al.*, "Fast distributed first-order methods," Ph.D. dissertation, Massachusetts Institute of Technology, 2012.
[3] J. C. Duchi, A. Agarwal, and M. J. Wainwright, "Dual averaging for distributed optimization: Convergence analysis and network scaling," *IEEE Transactions on Automatic Control*, vol. 57, no. 3, pp. 592–606, 2012.

[4] D. Jakovetic, J. Xavier, and J. M. Moura, "Fast distributed gradient methods," *IEEE Transactions on Automatic Control*, vol. 59, no. 5, pp. 1131–1146, 2014.
[5] 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.
[6] K. Yuan, Q. Ling, and W. Yin, "On the convergence of decentralized gradient descent," *SIAM Journal on Optimization*, vol. 26, no. 3, pp. 1835–1854, 2016.