非平衡有向网络的完全分布式凸优化

时侠圣[1,2], 林志赟[3], 王雪松[1,2†], 董世建[1,2]

(1. 中国矿业大学 地下空间智能控制教育部工程研究中心, 江苏 徐州 221116;

2. 中国矿业大学 信息与控制工程学院, 江苏 徐州 221116; 3. 南方科技大学 电子与电气工程系, 广东 深圳 518055)

摘要: 分布式凸优化问题的目的是如何以分布式方法最小化局部智能体成本函数和, 而现有分布式算法的控制步长选取依赖于系统智能体个数、伴随矩阵等全局性信息, 有悖于分布式算法的初衷. 针对此问题, 提出一种基于非平衡有向网络的完全分布式凸优化算法(FDCOA). 基于多智能体一致性理论和梯度跟踪技术, 设计了一种非负余量迭代策略, 使得FDCOA的控制步长收敛范围仅与智能体局部信息相关, 进而实现控制步长的分布式设置. 进一步分析了FDCOA在固定强连通和时变强连通网络情形下的收敛性. 仿真结果表明本文构建的分布式控制步长选取方法对FDCOA在有向非平衡下的分布式凸优化问题是有效的.

关键词: 分布式凸优化; 非平衡有向网络; 非负余量; 分布式步长

引用格式: 时侠圣, 林志赟, 王雪松, 等. 非平衡有向网络的完全分布式凸优化. 控制理论与应用, 2022, 39(6): 1071 – 1078

DOI: 10.7641/CTA.2021.10389

A fully distributed convex optimization algorithm over the unbalanced directed network

SHI Xia-sheng[1,2], LIN Zhi-yun[3], WANG Xue-song[1,2†], DONG Shi-jian[1,2]

(1. Engineering Research Center of Intelligent Control for Underground Space, Ministry of Education,
China University of Mining and Technology, Xuzhou Jiangsu 221116, China;
2. School of Information and Control Engineering, China University of Mining and Technology, Xuzhou Jiangsu 221116, China;
3. Department of Electrical and Electronic Engineering,
Southern University of Science and Technology, Shenzhen Guangdong 518055, China)

Abstract: The aim of the distributed convex optimization problem is how to minimize the sum of all local agent cost functions, and however, the control step-size of the existing distributed algorithms is related to the global information, such as agent numbers of system, adjacency matrix, which is contracted to the distributed algorithm. For solving this problem, a fully distributed convex optimization algorithm (FDCOA) is proposed over the unbalanced directed network. Based on the multi-agent consensus theory and gradient tracking technology, a non-negative surplus iteration scheme is designed to make the convergence range of the control step-size only related to the local information of each agent, and then to realize the uncoordinated and distributed setting of the control step-size. Further, the convergence analysis of the FDCOA is given for both fixed and time-varying strongly connected digraphs. The experimental results show that the designed distributed selection method of the control step-size is effective for the application of FDCOA to the distributed convex optimization problems under an unbalanced directed network.

Key words: distributed convex optimization; unbalanced directed network; non-negative surplus; distributed step-size

Citation: SHI Xiasheng, LIN Zhiyun, WANG Xuesong, et al. A fully distributed convex optimization algorithm over the unbalanced directed network. *Control Theory & Applications*, 2022, 39(6): 1071 – 1078

1 引言

5G通信在4G通信的基础上拥有更高的带宽和更快的数据传输速率, 随着5G通信的出现以及大规模应用, 5G通信技术的发展进一步推动了物联网、自动驾驶汽车和智能设备等新兴技术的发展和普及. 在这些应用中, 如何从各个分布式设备收集的数据中学习隐藏的参数是很有意义的事[1]. 这其中有一类问题可以定义为分类、回归或风险最小, 其核心是一个简单的

收稿日期: 2021−05−10; 录用日期: 2021−10−09.

†通信作者. E-mail: wangxuesongcumt@163.com; Tel.: +86 516-83590898.

本文责任编委: 梅生伟.

国家自然科学基金项目(62173118), 江苏省自然科学基金项目(BK20210492, BK20210493), 中央高校基本科研业务费青年科技基金项目(2021QN1052)资助.

Supported by the National Natural Science Foundation of China (62173118), the Natural Foundation of Jiangsu Province (BK20210492, BK20210493) and the Fundamental Research Funds for the Central Universities (2021QN1052).

成本和优化问题. 庞大的数据和隐私问题限制了数据共享, 因此, 局部和分布式优化问题得到了广泛的关注[2]. 而分布式多智能体技术的发展也促进了该问题的研究与发展. 在基于多智能体系统的分布式优化问题中, 各智能体协同最小化局部成本函数和. 这类优化问题在很多应用中得到关注[3–5], 比如分布式学习[6]、源定位[7]和编队控制[8]等.

近年来, 许多分布式优化算法被设计出来. 包括连续时间算法[9–14]和离散时间算法[15–16]. 连续时间算法要求所依赖智能体可连续执行微分方程, 例如机器人等. 而离散时间算法则是一种迭代策略, 更适应于计算机芯片等运行主体智能体, 应用较为广泛. 后续本文以离散迭代策略为目标, 阐述该领域最新研究成果. 分布式优化领域中最早的设计方法都是基于梯度下降[17]. 该算法也被广泛推广到受约束分布式优化问题[18]. 该类算法计算简单且直观, 但是衰减步长容易导致算法收敛速度降低. 此外, 基于拉格朗日对偶法的分布式凸优化算法也得到了研究, 进而引出了分布式对偶分解算法[19]和分布式交替乘子算法[20]. 相比于衰减步长梯度下降法, 该类算法虽然计算较复杂, 但收敛速度更快, 且可实现线性收敛. 为了同时实现算法的计算简单和快速收敛, 许多快速算法被提出, 比如分布式Nesterov加速算法[21–22]和分布式一阶精确算法[23]. 其中文献[22–23]通过梯度跟踪技术实现算法的固定步长控制.

注意到上述算法都是建立在无向网络基础上, 即若系统中智能体i能够发送信息给智能体j, 那么智能体j也能发送信息给智能体i. 如文献[24]所述, 上述算法的实现都需要构建一系列双随机矩阵. 然而在一些实际应用中, 此类通信机制是无法满足的. 例如拥有不同能量等级的智能体其通信能力是有区别的, 进而导致信息的单向传播. 在有向网络中, 双随机矩阵的特性(简称为平衡有向网络)一般难以满足, 因此接下来本文主要分析非平衡有向网络下的研究成果并做进一步的算法改进. 为了解决有向网络下的分布式优化问题, 有部分学者将Push-sum (推和)方法引入进来, 提出了一类分布式优化算法[25–26], 该方法的基本思想是通过动态跟踪伴随矩阵1特征根所对应的特征向量, 并利用此向量调整单向通信网络的不平衡性, 引导算法收敛到凸优化问题的最优解. 其中伴随矩阵只需确保列随机或者行随机. 后续很多学者基于此提出了加速算法[27–30], 其中文献[27]要求伴随矩阵行随机、文献[28]要求伴随矩阵列随机, 而文献[29–30]则分别是在文献[27]的基础上增加类Nesterov和重力球与Nesterov双动量项进一步提高算法收敛速度. 另一类算法则是通过同时构造一个列随机矩阵和一个行随机矩阵[31–33], 其中行随机矩阵确保了算法的一致性, 而列随机性矩阵则确保了算法的最优性. 文献[31]和文献[32]分别论证了固定强连通和时变强连通网络情形下的收敛性, 而文献[33]则是在文献[31]的基础上增加重力球动量项进一步提高算法的收敛速度.

虽然有向网络下的分布式凸优化问题已建立了很多优秀的算法, 但是上述算法的控制步长都依赖于很多全局信息, 比如网络伴随矩阵的最小正特征根、网络智能体个数等. 此外, 上述文献很少涉及到时变网络拓扑. 为此, 受文献[31]启发, 本文拟研究一类完全分布式凸优化算法(fully distributed convex optimization algorithm, FDCOA), 实现控制步长的分布式设置, 即控制步长的收敛范围仅与智能体局部信息相关. 进一步的, 本文拟研究所设计算法在时变强连通网络下的有效性.

符号说明 $\mathbb{R}$表示实数集; 对于实数$a \in \mathbb{R}$, $[a]_- = \min\{a, 0\}$; 对于向量$x = [x_1 \ \cdots \ x_n] \in \mathbb{R}^n$, $\underline{m}(x) = \min\limits_i\{x_i\}$和$\overline{m}(x) = \max\limits_i\{x_i\}$分别表示其中的最小值和最大值. 对于向量$x, y \in \mathbb{R}^n$, $x \succeq y$表示对于每一项都有$x_i \geqslant y_i, i = 1, 2, \cdots, n$. 对于矩阵有类似定义. ln表示以自然数$e$为底的对数函数, 简称自然对数.

2 问题描述与基础图论

2.1 图论基础知识

一个含有n个智能体的有向网络可以由符号$\mathcal{G} = (\mathcal{V}, \mathcal{E})$表示. 其中$\mathcal{V} = \{1, 2, \cdots, n\}$表示网络智能体集, $\mathcal{E}$ 表示网络中智能体之间通信链路组成的集合. (j, i)或$j \to i$表明存在一条从智能体j到智能体i传输信息的有向边. 智能体i_0到智能体i_l之间存在通路即存在一组智能体且有$i_0 \to i_1 \to \cdots \to i_{l-1} \to i_l$. 也称智能体$i_0$到$i_l$是可到达的. 更进一步地, 如果有向图$\mathcal{G}$中所有智能体都是互相可到达的, 则称$\mathcal{G}$为强连通图. 此外, 定义智能体$i$的入和出邻居集分别为$N_i^{\text{in}} = \{j | (j, i) \in \mathcal{E}\}$和$N_i^{\text{out}} = \{j | (i, j) \in \mathcal{E}\}$. 此外, 默认智能体都能接收到自身的信息, 即$i \in N_i^{\text{in}}$和$i \in N_i^{\text{out}}$. 并令$|N_i^{\text{in}}|$和$|N_i^{\text{out}}|$分别表示智能体$i$的入度与出度.

假设 1 在本文所研究凸优化问题中, 智能体之间的通信网络为有向强连通的.

假设1是无领导者多智能体系统分布式优化问题的必要条件, 其保证了多智能体各局部信息的全局性流通, 确保全局最优解的获取.

2.2 问题描述

在分布式凸优化问题中, 针对每个智能体i, 存在一个本地决策变量$x_i \in \mathbb{R}$和一个凸成本函数$f_i(x_i) =: \mathbb{R} \to \mathbb{R}$. 该问题的目标是整个网络的成本函数之和最小, 且各智能体的决策变量达到一致. 所以, 分布式凸优化的数学模型如下:

$$
\begin{aligned}
&\min_{x_1, x_2, \cdots, x_n} \sum_{i=1}^{n} f_i(x_i),\\
&\text{s.t. } x_i = x_j, \ \forall i, j \in \mathcal{V},
\end{aligned} \tag{1}
$$

其中各智能体i只能获取自身的成本函数$f_i(x_i)$. 上述问题在很多应用中都可见到, 比如经典的源定位问题: $f_i(x_i)=(x_i-r_i)^2$, 其中x_i表示传感器i的局部估计值, r_i表示传感器到目标点的测量距离. 需要注意的是, 本文为了行文简洁, 将问题设定为一维情形. 但所设计结论可直接推广至一般情形, 在后续实验部分给出相应的案例仿真.

为了解决上述问题, 提出以下几个假设.

假设 2　针对每个智能体i, 假设其局部成本函数$f_i(x_i)$是严格凸的. 即满足条件: $(x-y)\ (\nabla f_i(x)-\nabla f_i(y))>0,\ \forall x,y\in\mathbb{R}$.

假设 3　针对每个智能体i, 假设其局部成本函数$f_i(x_i)$是κ_i-李普希茨连续的. 即满足条件: $|(\nabla f_i(x)-\nabla f_i(y))|\leqslant\kappa_i|x-y|,\ \forall x,y\in\mathbb{R}$.

假设2确保了优化问题(1)解的唯一性, 同时也有利于后续算法收敛性分析. 此假设要弱于文献[22,26–33]的强凸函数假设. 假设3是对局部成本函数的光滑度描述, 即函数的梯度不会变化太突兀, 其中κ_i为光滑性量度. 此条件对很多基于梯度下降的分布式凸优化算法的收敛性分析很重要, 比如文献[16–17,19–23,25–33]. 此外, 如定理1所示, 光滑性量度κ_i影响算法控制参数的选取范围.

3　算法设计

为了实现控制步长的分布式设置, 基于梯度跟踪技术, 本文调整智能体决策变量x_i的获取方式, 实现余量变量的非负性. 具体算法1设计如下:

$$x_{i,k+1}=x_{i,k}+[\sum_{j=1}^{n}a_{ij}(x_{j,k}-x_{i,k})]_-+\alpha_i s_{i,k},\tag{2a}$$

$$s_{i,k+1}=\sum_{j=1}^{n}b_{ij}s_{j,k}-(\nabla f_i(x_{i,k+1})-\nabla f_i(x_{i,k})),\tag{2b}$$

式中: 正数α_i是控制参数, 伴随矩阵定义为

$$A:=[a_{ij}]=\frac{1}{|N_i^{\rm in}|},\ B:=[b_{ij}]=\frac{1}{|N_j^{\rm out}|},$$

若存在$(j,i)\in\mathcal{E}$, 否则$a_{ij}=0, b_{ij}=0$;系统初始状态满足$\sum_{i=1}^{n}(s_{i,0}+\nabla f_i(x_{i,0}))=0$. 此外, 令$x(k)=[x_{1,k}\ \ x_{2,k}\ \ \cdots\ \ x_{n,k}]$, $s(k)=[s_{1,k}\ \ s_{2,k}\ \ \cdots\ \ s_{n,k}]$表示其向量形式. 算法1的物理意义如下: 首先基于多智能体一致性方法, 式(2b)实现智能体梯度信息跟踪, 即$s_{i,k}\to -\dfrac{1}{n}\sum_{i=1}^{n}\nabla f_i(x_{i,k})$. 其次, 基于多智能体一致性方法和梯度下降法, 令智能体状态$x_{i,k}$沿着余量方向, 协同迭代至凸优化问题的最优解.

本文所设计算法1与文献[31]所设计算法都解决了有向网络下的分布式凸优化问题(1), 但区别点如下: 1)文献[31]利用线性项$\sum_{j=1}^{n}a_{ij}(x_{j,k}-x_{i,k})$, 而本文算法1使用非线性项$[\sum_{j=1}^{n}a_{ij}(x_{j,k}-x_{i,k})]_-$, 进而实现余量$s_i$的非负; 2)文献[31]使用共同的控制参数$\alpha_i$, 且其收敛范围与网络总智能体个数、伴随矩阵关于特征根1的特征向量、和全局目标函数的光滑性量度等全局性信息相关. 而本文所设计算法1控制参数仅与智能体局部邻居智能体个数和智能体局部目标函数的光滑性量度等局部性信息相关.

在给出正式的收敛性分析之前, 先给出算法1中一个很重要的特性, 即余量–梯度和不变性. 从式(2b)可得

$$\begin{aligned}&\sum_{i=1}^{n}(s_{i,k+1}+\nabla f_i(x_{i,k+1}))=\\&\sum_{i=1}^{n}\sum_{j=1}^{n}b_{ij}s_{j,k}+\sum_{i=1}^{n}\nabla f_i(x_{i,k})=\\&\sum_{j=1}^{n}s_{j,k}+\sum_{i=1}^{n}\nabla f_i(x_{i,k})=\\&\sum_{i=1}^{n}s_{i,k}+\sum_{i=1}^{n}\nabla f_i(x_{i,k})=\\&\quad\vdots\\&\sum_{i=1}^{n}s_{i,0}+\sum_{i=1}^{n}\nabla f_i(x_{i,0})=0,\end{aligned}\tag{3}$$

其中第2个等式的成立利用了伴随矩阵B的列随机性, 最后一个等式考虑了算法的初值约束. 可得结论

$$\sum_{i=1}^{n}(s_{i,k+1}+\nabla f_i(x_{i,k+1}))=0,\ \forall,\ k\geqslant 0.$$

在算法1, 虽然对算法初始值有约束, 但是不管其形式多么复杂, 只要确保局部目标函数$f_i(x_i)$是凸函数, 即假设(2)成立. 则此局部解都可以通过离散迭代策略$x_{i,k+1}=x_{i,k}-\beta\nabla f_i(x_{i,k}), \beta>0$迭代获得. 此时可令智能体初始状态满足$\nabla f_i(x_i(0))=0$, $s_i(0)=0$即可.

4　算法收敛性分析

为了完成算法1的收敛性分析, 接下来首先引入几个重要的引理.

引理 1　若假设1和3成立, 则对任意$i\in\mathcal{V}, k\geqslant 0$和算法1, 有结论

$$0\leqslant s_i(k)\leqslant\frac{\underline{m}(x(k+\varDelta+1))-\underline{m}(x(k+\varDelta))}{\mu^{\varDelta+1}},\tag{4}$$

式中: $\mu=\min\limits_i\{\alpha_i,a_{ii},b_{ii}-\kappa_i\alpha_i\}, \varDelta=n-1$.

证　从式(2a)可知

$$x_{i,k+1}-x_{i,k}\leqslant\alpha_i s_{i,k}.\tag{5}$$

结合局部目标函数的李普希茨性质, 有结论

$$\nabla f_i(x_{i,k+1}) - \nabla f_i(x_{i,k}) \leqslant \kappa_i\alpha_i s_{i,k}. \tag{6}$$

将上述结果代入式(2b), 可得

$$s_{i,k+1} \geqslant (b_{ii} - \kappa_i\alpha_i)s_{i,k} + \sum_{j=1,j\neq i}^{n} b_{ij}s_{j,k}. \tag{7}$$

考虑到余量满足$s_{i,0}=0$, 则余量s_i的非负性得以成立. 进而由式(7)可推导出$s(k+1)\succeq\mu[I_n+S]s(k)$, 其中矩阵$S$定义为$S_{ij}=1$若存在通信链路$(j,i)\in\mathcal{E}$. 所以对于任意的$k>k_0\geqslant 0$, 有

$$\begin{aligned} s(k+1) \succeq \mu^{k-k_0}(I_n+S)^{k-k_0}s(k_0) \succeq \\ \mu^{k-k_0}(I_n+(k-k_0)S)s(k_0). \end{aligned} \tag{8}$$

那么对于任意的$i\in\mathcal{V}$, 下式成立:

$$s_{i,k} \geqslant \mu^{k-k_0}s_{i,k_0}. \tag{9}$$

由假设1可知, 对于任意两智能体i,j, 存在一条有向路$j\to l_m\to\cdots\to l_1\to i$, 其中路径长度$m$满足$m\in\{1,2,\cdots,n-1\}$. 结合式(7), 则有

$$s_{i,k} \geqslant \mu^{m}s_{j,k-m}. \tag{10}$$

将式(9)–(10)结合, 可得

$$s_{i,k+\varDelta} \geqslant \mu^{\varDelta}s_{j,k},\ \forall i,j\in\mathcal{V}. \tag{11}$$

由于$x_{i,k}+[\sum\limits_{j=1}^{n}a_{ij}(x_{j,k}-x_{i,k})]_-$是$x_{i,k}$和$x_{j,k}, j\in N_i^{\text{in}}$的组合, 所以有$x_{i,k}+[\sum\limits_{j=1}^{n}a_{ij}(x_{j,k}-x_{i,k})]_-\geqslant x_{j,k}$, $j\in N_i^{\text{in}}$, 所以有结论

$$\begin{aligned} x_{i,k+1} \geqslant \min_{j\in N_i^{\text{in}}\cup\{i\}} x_{j,k}+\alpha_i s_{i,k} \geqslant \\ \underline{m}(x(k))+\mu s_{i,k},\ \forall i\in\mathcal{V}. \end{aligned} \tag{12}$$

令上式的k为$k+\varDelta$, 且存在一个智能体i_1, 使得$x_{i_1,k+\varDelta+1}=\underline{m}(x(k+\varDelta+1))$. 代入上式可得

$$s_{i_1,k+\varDelta} \leqslant \frac{\underline{m}(x(k+\varDelta+1))-\underline{m}(x(k+\varDelta))}{\mu}. \tag{13}$$

结合式(11)中i,j的任意性, 令$i=i_1$, 则有结论

$$s_{j,k} \leqslant \frac{\underline{m}(x(k+\varDelta+1))-\underline{m}(x(k+\varDelta))}{\mu^{\varDelta+1}}. \tag{14}$$

继续令$j=i$, 引理中结论可得证. 证毕.

上述引理给出了余量$s(k)$的动态上界. 为了表明$\lim\limits_{k\to\infty}s(k)=0$, 则只需要证明序列$\underline{m}(x(k))$存在极限. 这就等价于去证明$\underline{m}(x(k))$是有界的, 且不会振荡.

引理 2 序列$\underline{m}(x(k))$非降, 且$\underline{m}(x(k))\leqslant x^*$, $\forall k\geqslant 0$, 其中x^*表示问题(1)的最优解.

证 由式(12)可得$x_{i,k+1}\geqslant\underline{m}(x(k))$, 这就表明序列$\underline{m}(x(k))$关于迭代次数$k$是非降的. 接下来用反证法分析, 由反证法可知存在$k\geqslant 0$使得$\underline{m}(x(k))>x^*$. 由于目标函数是凸函数, 所以有结论$\sum\limits_{i=1}^{n}\nabla f_i(x_{i,k})>\sum\limits_{i=1}^{n}\nabla f_i(x^*)=0$. 结合算法的余量–梯度和不变性, 可得$\sum\limits_{i=1}^{n}s_{i,k}<0$, 此结果与余量的非负性矛盾. 则结论得证. 证毕.

记序列$\underline{m}(x(k))$的上确界为$\sigma=\sup(\underline{m}(x(k)))$, 结合引理2, 可得

$$\lim_{k\to\infty}\underline{m}(x(k))=\sigma. \tag{15}$$

引理 3 若假设2成立, 算法1的唯一稳定点是$x=x^*1_n, s=0$.

证 首先, 将上述结论$x_{i,k}=x^*, s_{i,k}=0$代入算法1, 可得$x_{i,k+1}=x_{i,k}, s_{i,k+1}=s_{i,k}$, 这就表明上述结论是算法1 的稳定点.

接下来分析其唯一性. 假设算法存在另外一组稳定点, 即智能体状态不发生变化, 则由引理2可知, $s_i(k)=0$. 结合式(2a), 可得$[\sum\limits_{j=1}^{n}a_{ij}(x_{j,k}-x_{i,k})]_-=0$. 则存在一智能体$i_1$, 使得$x_{i_1,k}=\overline{m}(x(k))$, 那么其所有的邻居智能体状态必满足$x_{i,k}=x_{i_1,k}$, 否则$[\sum\limits_{j=1}^{n}a_{ij}(x_{j,k}-x_{i,k})]_-\neq 0$. 进一步地, 由于通信拓扑的强连通性可知, $x_{i,k}=x_{j,k},\forall i,j\in\mathcal{V}$. 结合余量-梯度和不变性和目标函数的凸性, 可知$x_i=x^*,\forall i\in\mathcal{V}$. 证毕.

定理 1 若假设1–3成立, 则算法1收敛至问题(1)的最优解. 不同智能体拥有不同的控制参数, 且控制参数满足$0<\alpha_i<\dfrac{b_{ii}}{\kappa_i}$.

证 由引理1和式(15)可知$\lim\limits_{k\to\infty}s(k)=0$. 在引理3中, 证明了系统稳定点的唯一性. 接下来只需要证明对任意的智能体i, 其状态$x_{i,k}$极限存在即可. 这就等价于证明$x_{i,k}$是有界的, 且不会振荡.

对于任意智能体i和$k\geqslant k_0\geqslant 0$, 从式(2a)可得

$$x_{i,k} \leqslant x_{i,k_0}+\alpha_i\sum_{t=k_0}^{k-1}s_{i,t} \leqslant x_{i,k_0}+\alpha_i\sum_{t=k_0}^{\infty}s_{i,t}. \tag{16}$$

令$h_{i,k_0}=\alpha_i\sum\limits_{t=k_0}^{\infty}s_{i,t}$, 则上式可重写为

$$x_{i,k}\leqslant x_{i,k_0}+h_{i,k_0}. \tag{17}$$

接下来的分析分成3部分: 1) h_{i,k_0}是有界的; 2) $x_{i,k}$是有界的; 3) 序列$\{x_{i,k}\}$不会振荡.

步骤 1 h_{i,k_0}是有界的, 且$\lim\limits_{k_0\to\infty}h_{i,k_0}=0$, $\forall i\in\mathcal{V}$. 很显然, 由余量变量的非负性可得$h_{i,k_0}\geqslant 0$. 此外,

由引理1可知

$$h_{i,k_0} \leqslant \alpha_i \sum_{t=k_0}^{\infty} \frac{\underline{m}(x(t+\Delta+1)) - \underline{m}(x(t+\Delta))}{\mu^{\Delta}} = \alpha_i \frac{\lim\limits_{t\to\infty} \underline{m}(x(t+\Delta+1)) - \underline{m}(x(k_0+\Delta))}{\mu^{\Delta}} = \alpha_i \frac{\sigma - \underline{m}(x(k_0+\Delta))}{\mu^{\Delta}}. \tag{18}$$

结合序列$\{\underline{m}(x(k))\}$的非降性, 对上式两边k_0同时取极限, 结论可得.

步骤 2 对任意智能体i, 序列$\{x_{i,k}\}$是有界的. 由于$\underline{m}(x(k)) \leqslant x_{i,k} \leqslant x_{i,k_0} + h_{i,k_0}$, 且$\{\underline{m}(x(k))\}$非降, h_{i,k_0}有界, 则可直接得出结论: $x_{i,k}$是有界的.

步骤 3 对于任意的$i \in \mathcal{V}$, $\{x_{i,k}\}$不会发生振荡. 反证法, 假设存在一智能体i, $\{x_{i,k}\}$发生振荡现象. 则可以构造两个收敛到不同值的子序列$\{x_{i,k_j}\}$和$\{x_{i,k_l}\}$, 可令$\lim\limits_{k_j\to\infty} x_{i,k_j} = a$, $\lim\limits_{k_l\to\infty} x_{i,k_j} = b$. 不失一般性, 设$a > b$. 那么对于任意的$k_l$, 都可找到$k_j > k_l$. 从式(17)可得

$$\lim_{k_l\to\infty} h_{i,k_l} \geqslant \lim_{k_l\to\infty} (x_{i,k_j} - x_{k_l}) = a - b > 0. \tag{19}$$

此结果与结论$\lim\limits_{k\to\infty} h_{i,k} = 0$矛盾. 由反证法可知序列$\{x_{i,k}\}$不会发生振荡.

综合以上3步, 利用引理3, 可知$\lim\limits_{k\to\infty} x_{i,k} = x^*$, $\forall i \in \mathcal{V}$. 则定理得证. 证毕.

从定理1可以看出, 算法1中控制参数α_i可以分布式获取, 且仅与自身的局部信息相关. 为了展示本文算法控制参数的便捷性, 将文献[23, 31]控制参数α_i的收敛范围给出$0 < \alpha_i < \dfrac{2}{n\kappa\pi_r^{\mathrm{T}}\pi_c}$和$0 < \alpha_i < \dfrac{2\rho}{\kappa}$, 其中$\kappa = \max\limits_i \kappa_i$, π_r, π_c分别是伴随矩阵A关于1特征根所对应的右和左特征向量, ρ是伴随矩阵的最小正特征根. 由此可以看出, 文献[23, 31]中算法控制参数选取与系统智能体个数、伴随矩阵等全局性信息有关, 且随着系统智能体个数的增多, 其收敛范围减小.

由上述分析可知, 定理1证明了算法1在固定有向强连通网络下的收敛性. 接下来继续探讨该算法在时变有向强连通网络下的收敛性分析. 首先给出相关的网络连通性假设.

假设 4 时变通信网络$\mathcal{G}(k)$是周期强连通的, 即存在正整数$K > 0$, 使得$\mathcal{G}([k_0, k_0+K])$是强连通的, 其中$\mathcal{G}([k_0, k_0+K]) := (\mathcal{V}, \cup_{k\in[k_0,k_0+K]}\mathcal{E}(k))$.

为了解决时变网络下的分布式凸优化问题, 算法2中的通信链路增益也需要动态改变, 具体的算法2设计如下:

$$x_{i,k+1} = x_{i,k} + [\sum_{j=1}^{n} a_{ij,k}(x_{j,k} - x_{i,k})]_- + \alpha_i s_{i,k}, \tag{20a}$$

$$s_{i,k+1} = \sum_{j=1}^{n} b_{ij,k} s_{j,k} - (\nabla f_i(x_{i,k+1}) - \nabla f_i(x_{i,k})), \tag{20b}$$

式中:

$$a_{ij,k} = \frac{1}{|N_i^{\mathrm{in}}(k)|},\ b_{ij,k} = \frac{1}{|N_j^{\mathrm{out}}(k)|},$$

若存在$(j, i) \in \mathcal{E}$, 否则$a_{ij,k} = 0, b_{ij,k} = 0$; 系统初始状态满足$s_{i,0} = \nabla f_i(x_{i,0}) = 0$; 正数$\alpha_i$是控制参数.

推论 1 若假设1, 2和4成立, 则算法2收敛至问题(1)的最优解. 不同智能体拥有不同的控制参数, 且控制参数满足$0 < \alpha_i < \dfrac{b_{ii,k}}{\kappa_i}$.

证 由算法2可以看出, 只需对引理2进行调整即可. 由假设4可知存在正整数$K > 0$, 使得$\mathcal{G}([k_0, k_0+K])$满足强连通. 所以引理2的$\Delta$调整为$\Delta = K(n-1)$. 由于后续证明是相似的, 故在此不再赘述.

证毕.

5 案例分析

为了验证所提算法的有效性, 接下来本文利用MATLAB演示3个案例的仿真效果.

案例 1 首先考虑由4个多智能体组成的有向网络分布式凸优化问题[16], 每个智能体的局部成本函数定义为

$$f(x_1) = x_1^2 - 2x_1 + \cos x_1 + 3,$$
$$f(x_2) = x_2^2 - 5x_2 + \exp(-0.1x_2) - 1,$$
$$f(x_3) = x_3^2 - 3x_3 - 0.5\sin x_3 - 3,$$
$$f(x_4) = x_4^2 + 2x_4 - 3.$$

智能体间的通信链路如图1所示. 由算法1可知, 智能体i只需要将自身信息$x_{i,k}$和$b_{ji}s_{i,k}$发送给其邻居智能体j.

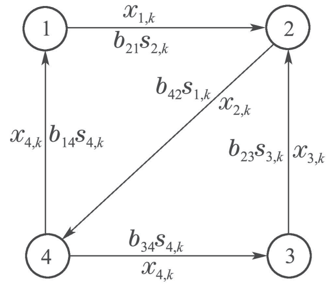

图 1 4智能体系统通信网络

Fig. 1 The communication network for four agents

如算法1所示, 首先获取各局部成本函数驻点$x(0)$

$= [1.4987, 2.5388, 1.5141, -1]$, 且案例问题的最优解为$x^* = 1.1508$. 令$\alpha_i = 0.95\frac{b_{ii}}{\kappa_i}, \forall i \in \mathcal{V}$, 算法1运行轨迹如图2和3所示. 从图2可看出, 各智能体协同收敛至最优解x^*. 从图3可以看出, 算法1所设计的余量控制策略可确保变量s_i的非负性, 且最终都趋向于零. 此外, 从前面的控制步长α_i的选取可以看出, 本文所设计算法1实现了控制步长的分布式选取, 即只需利用各智能体局部信息, 克服了同类分布式算法中控制步长选取困难的问题. 进一步, 为了验证算法2的有效性, 基于上述数据, 考虑如图4所示的通信网络. 将算法1在图1与算法2在图4下的误差

$$e(k) = \ln(\frac{1}{2}\sum_{i=1}^{n}(x_i - x^*)^2)$$

展示在图5中. 从图5可以看出, 本文所设计算法在时变强连通网络下亦是有效的, 且固定通信网络下收敛速度相对更快一点.

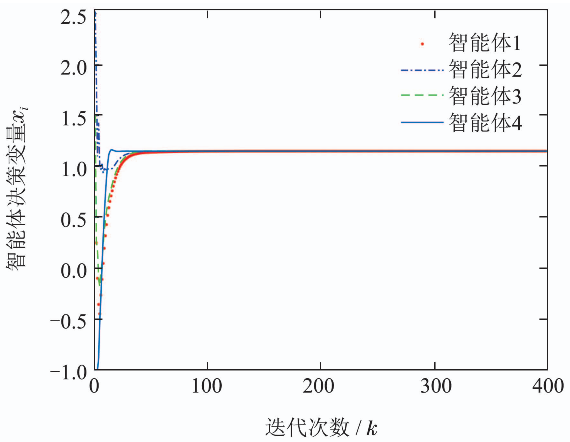

图 2 案例1中各智能体状态x_i轨迹图

Fig. 2 The trajectories of state variable x_i for each agent in case 1

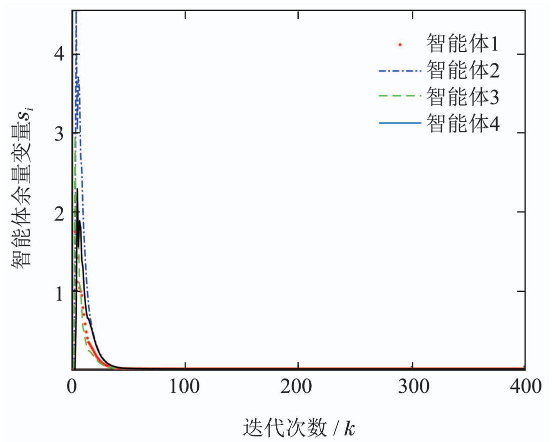

图 3 案例1中各智能体余量s_i轨迹图

Fig. 3 The trajectories of surplus variable s_i for each agent in case 1

案例 2 验证本文所设计算法1在大规模网络下的有效性. 以文献[32]的分布式最小二乘问题为例, 各智能体协同寻找最小二乘问题的解, 其中局部损失函数定义为$f_i(x) = \frac{1}{2}\|H_i x - b_i\|_2^2$, 在此案例中, 令智能体个数为$n = 500$, $H_i \in \mathbb{R}^{2\times 1}$. 智能体间的通信拓扑如图6所示. 算法1决策变量运行轨迹如图7所示, 各智能体协同收敛到最优解. 同案例1, 算法运行误差轨迹图如图8 所示. 虽然文献[32]矩阵奇异摄动法证明了所设计算法在时变网络下的收敛性, 但没有给出控制步长的收敛范围, 且各智能体拥有相同的控制步长, 不利于分布式系统的实际应用.

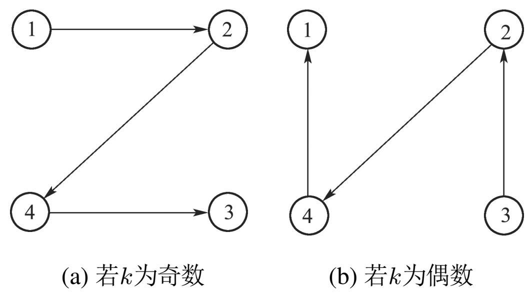

图 4 4智能体时变强连通网络

Fig. 4 The communication network for four agents

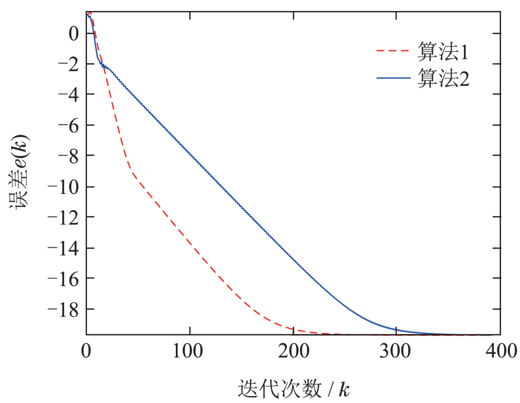

图 5 案例1中两种算法误差e_i轨迹图

Fig. 5 The convergence comparison between the two proposed algorithms in case 1

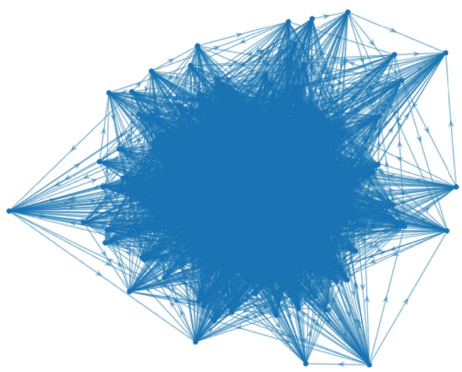

图 6 案例2中500智能体系统通信网络

Fig. 6 The communication network for 500 agents in case 2

案例 3　为了验证本文所提算法在一般维度情形下分布式凸优化问题的有效性, 如文献[34]所示的二次布局问题(quadratic placement problem), 包含5个固定点和4个自由点, 其中固定点分别为$t_1=(-1,-1)^{\mathrm{T}}$, $t_2=(-1,0)^{\mathrm{T}}$, $t_3=(0,1)^{\mathrm{T}}$, $t_4=(1,0)^{\mathrm{T}}$, $t_5=(1,-1)^{\mathrm{T}}$, 自由点的初始状态为$x_1(0)=(-\frac{1}{3},-\frac{2}{3})^{\mathrm{T}}$, $x_2(0)=(1,-0.5)^{\mathrm{T}}$, $x_3(0)=(0.5,0.5)^{\mathrm{T}}$, $x_4(0)=(-0.5,0.5)^{\mathrm{T}}$. 自由节点的通信网络如图1所示. 此外, 自由点的目标函数设置为

$$\begin{cases} f_1(x)=\|x-t_1\|^2+\|x-t_2\|^2+\|x-t_5\|^2,\\ f_2(x)=\|x-t_4\|^2+\|x-t_5\|^2,\\ f_3(x)=\|x-t_3\|^2+\|x-t_4\|^2,\\ f_4(x)=\|x-t_2\|^2+\|x-t_3\|^2. \end{cases} \tag{21}$$

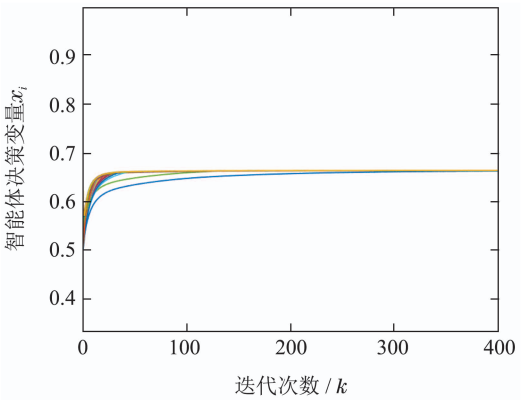

图 7　案例2中各智能体状态x_i轨迹图

Fig. 7　The trajectories of state variable x_i for each agent in case 2

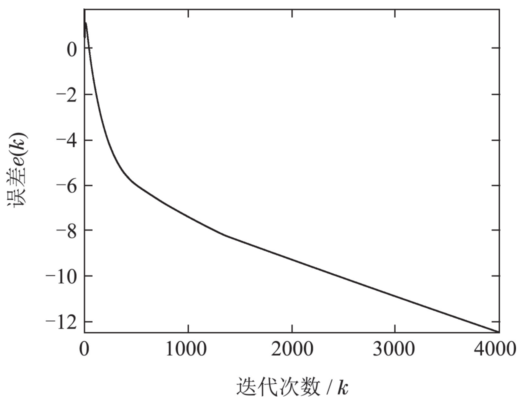

图 8　案例2中误差$e(k)$轨迹图

Fig. 8　The trajectory of error $e(k)$ in case 2

可以得出$\kappa_1=6$, $\kappa_2=\kappa_3=\kappa_4=4$. 此外上述优化问题的理论最优解为$x^*=(\frac{1}{9},-\frac{1}{9})$, 自由点的状态变化如图9所示. 可以看出, 本文所设计算法对一般维度情形分布式优化问题亦是有效的.

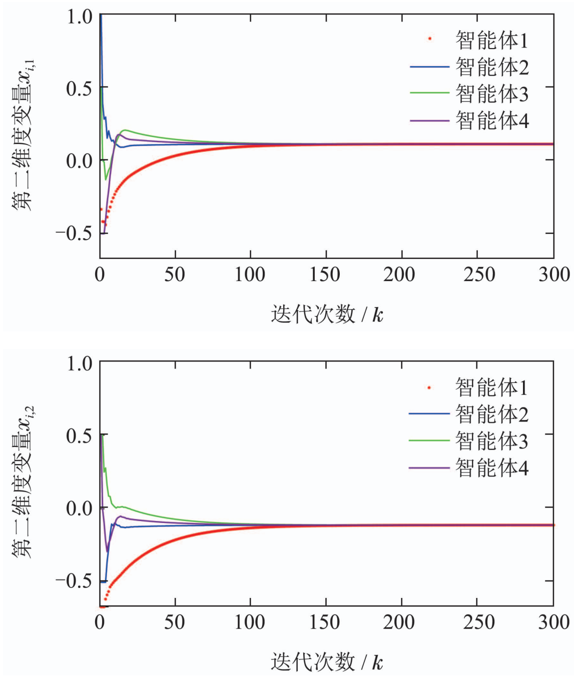

图 9　案例3中各智能体状态x_i轨迹图

Fig. 9　The trajectories of state variable x_i for each agent in case 3

6　结论

本文针对非平衡有向强联通网络下的分布式凸优化问题, 提出了一种完全分布式优化算法. 在算法设计中, 每个智能体只能获取自身的成本函数信息. 通过调整智能体决策变量的迭代策略, 实现了余量变量的非负性控制和控制步长的分布式设置. 其次, 本算法对时变强连通网络亦是有效的. 仿真结果表明, 本文所设计算法可以有效解决有向网络下的凸优化问题.

参考文献:

[1] RAPPAPORT T S, SUN S, MAYZUS R, et al. Millimeter wave mobile communications for 5G cellular: It will work! *IEEE Access*, 2013, 1: 335 – 349.

[2] SHAHRAMPOUR S, RAKHLIN S, JADBABAIE A. Online learning of dynamic parameters in social networks. *Proceeding of the 26th Advances in Neural Information Processing Systtems*. Lake Tahoe: ACM, 2013: 2013 – 2021.

[3] YANG Tao, CHAI Tianyou. Research status and prospects of distributed collabotative optimization. *Scientia Sinica Technologica*, 2020, 50(11): 1414 – 1425.
(杨涛, 柴天佑. 分布式协同优化的研究现状与展望. 中国科学: 技术科学, 2020, 50(11): 1414 – 1425.)

[4] XIN R, PU S, NEDIC A, et al. A general framework for decentralized optimization with first-order methods. *Proceeding of the IEEE*, 2020, 108(11): 1869 – 1889.

[5] WANG Long, LU Kaihong, GUAN Yongqiang. Distributed optimization via multi-agent systems. *Control Theory & Applications*, 2019, 36(11): 1820 – 1833.
(王龙, 卢开红, 关永强. 分布式优化的多智能体方法. 控制理论与应用, 2019, 36(11): 1820 – 1833.)

[6] CEVHER V, BECKER S, SCHMIDT M. Convex optimization for big data: Scalable, randomized, and parallel algorithms for big data anaytics. *IEEE Signal Processing Magazine*, 2014, 31(5): 32 – 43.

[7] SAFAVI S, KHAN U A. An opportunistic algorithm for localization in mobile networks. *IEEE Transactions on Robotics*, 2017, 33(4): 875 – 888.

[8] REN W, BEARD R W, ATKINS E M. Information consensus in multi-vehicle cooperative control. *IEEE Control Systems Magazine*, 2007, 27(2): 71 – 82.

[9] SHI X, CAO J, WEN G, et al. Finite-time consensus of opnion dynamics and its applications to distributed optimization over digraph. *IEEE Transactions on Cybernetics*, 2018, 49(10): 3767 – 3779.

[10] LI Z, WU Z, LI Z, et al. Distributed optimal coordination for heterogenous linear multiagent systems with event-triggered mechanisms. *IEEE Transactions on Automatic Control*, 2020, 65(4): 1763 – 1770.

[11] ZHU Y, YU W, WEN G, et al. Continuous-time distributed subgradient algorithm for convex optimization with general constraints. *IEEE Transactions on Automatic Control*, 2019, 64(4): 1694 – 1701.

[12] CHEN R, YANG T, CHAI T. Distributed accelerated optimization algorithms: Insights from an ODE. *Sicence China Technological Sciences*, 2020, 63(9): 1647 – 1655.

[13] YANG Tao, XU Lei, YI Xinlei, et al. Event-triggered distributed optimization algorithms. *Acta Automatica Sinica*, 2021, DOI: 10.16383/j.aas.c200838.
(杨涛, 徐磊, 易新蕾, 等. 基于事件触发的分布式优化算法. 自动化学报, 2021, DOI: 10.16383/j.aas.c200838.)

[14] SHI Xiasheng, YANG Tao, LIN Zhiyun, et al. Distributed resource allocation algorithm for second-order multi-agent systems in continuous-time. *Acta Automatica Sinica*, 2021, 47(8): 2050 – 2060.
(时侠圣, 杨涛, 林志赟, 等. 基于连续时间的二阶多智能体分布式资源分配算法. 自动化学报, 2021, 47(8): 2050 – 2060.)

[15] LU K, JING G, WANG L. Online distributed optimization with strongly pseudoconvex-sum cost functions. *IEEE Transactions on Automatic Control*, 2020, 65(1): 426 – 433.

[16] WANG D, WANG D, WANG W, et al. A modified distributed optimization method for both continuous-time and discrete-time multiagent systems. *Neurocomputing*, 2018, 275: 725 – 732.

[17] NEDIC A, OZDAGLAR A. Distributed subgradient methods for multi-agent optimization. *IEEE Transactions on Automatic Control*, 2009, 54(1): 48 – 61.

[18] NEDIC A, OZDAGLAR A, PARRILO P A. Constrained consensus and optimization in multi-ageny networks. *IEEE Transactions on Automatic Control*, 2010, 55(4): 922 – 938.

[19] TERELIUS H, TOPCU U, MURRAY R M. Decentralized multiagent optimization via dual decomposition. *IFAC Proceedings Volumes*, 2011, 44(1): 11245 – 11251.

[20] SHI W, LING Q, YUAN K, et al. On the linear convergence of the ADMM in decentralized consensus optimization. *IEEE Transactions on Signal Processing*, 2014, 62(7): 1750 – 1761.

[21] JAKOVETIC D, XAVIER J, MOURA J M F. Fast distributed gradient methods. *IEEE Transactions on Automatic Control*, 2014, 59(5): 1131 – 1146.

[22] QU G, LI N. Harnessing smoothness to accelerate distributed optimization. *IEEE Transactions on Control Network Systems*, 2018, 5(3): 1245 – 1260.

[23] SHI W, LING Q, WU G, et al. EXTRA: An exact first-order algorithm for decentalized consensus optimization. *SIAM Journal of Optimization*, 2015, 25(2): 944 – 966.

[24] XIAO Li, BAO Junjie, SHI Xi, et al. A primal-dual algorithm for solving distributed economic allocation problem over a directed unbalanced network. *Journal of Southwest University (Natural Science Edition)*, 2018, 40(11): 48 – 54.
(肖丽, 包骏杰, 石熙, 等. 非平衡有向网络上求解分布式经济分配问题的原始–对偶算法. 西南大学(自然科学版), 2018, 40(11): 48 – 54.)

[25] NEDIC A, OLSHEVSKY A. Distributed optimization over timevarying directed graphs. *IEEE Transactions on Automatic Control*, 2015, 60(3): 601 – 615.

[26] XI C, KHAN U A. DEXTRA: A fast algorithm for optimization over directed graphs. *IEEE Transactions on Automatic Control*, 2017, 62(10): 4980 – 4993.

[27] XI C, MAI V S, XIN R, et al. Linear convergence in optimization over directed graphs with row-stochastic matrices. *IEEE Transactions on Automatic Control*, 2018, 63(10): 3558 – 3565.

[28] XI C, XIN R, KHAN U A. ADD-OPT: Accelerated distributed directed optimization. *IEEE Transactions on Automatic Control*, 2018, 63(5): 1329 – 2339.

[29] LI H, CHENG H, WANG Z, et al. Distributed nesterov gradient and heavy-ball double accelerated asynchronous optimization. *IEEE Transactions on Nerual Network and Learning Systems*, 2020, DOI: 10.1109/TNNLS.2020.3027381.

[30] LV Q, LIAO X, LI H, et al. A nesterov-like gradient tracking algorithm for distributed optimization over directed networks. *IEEE Transactions on Systemts Man Cybernetics: Systems*, 2021, 51(10): 6258 – 6270.

[31] XIN R, KHAN U A. A linear algorithm for optimization over directed graphs with geometric convergence. *IEEE Control Systems Letters*, 2018, 2(3): 315 – 320.

[32] SAADATNIAKI F, XIN R, KHAN U A. Decentralized optimization over time-varying directed graphs with row and coloumn-stochastic matrices. *IEEE Transactions on Automatic Control*, 2020, 65(11): 4769 – 4780.

[33] XIN R, KHAN U A. Distributed heavy-ball: A generalization and acceleration of first-order methods with gradient tracking. *IEEE Transactions on Automatic Control*, 2020, 65(6): 2627 – 2633.

[34] MO L, HU H, YU Y, et al. Distributed optimization without boundedness of gradients for second-order multi-agent systems over unbalanced network. *Infomation Sciences*, 2021, 565: 177 – 195.

作者简介:

时侠圣 讲师, 目前研究方向为多智能体分布式优化、事件触发控制和固定时间收敛等, E-mail: shixiasheng@cumt.edu.cn;

林志赟 教授, 博士生导师, 俄罗斯工程院外籍院士, 目前研究方向为多智能体控制、优化及应用, E-mail: linzy@sustech.edu.cn;

王雪松 教授, 博士生导师, 目前研究方向为强化学习、宽度学习、迁移学习等人工智能方法及应用, E-mail: wangxuesongcumt@163.com;

董世建 副教授, 目前研究方向为非线性系统建模、控制与优化, Email: dsjggy@126.com.