



分布式一致性最优化的梯度算法与收敛分析

梁舒 彭开香

Distributed gradient-based consensus optimization algorithm and convergence analysis

LIANG Shu, PENG Kaixiang

引用本文:

梁舒, 彭开香. 分布式一致性最优化的梯度算法与收敛分析[J]. *工程科学学报*, 2020, 42(4): 434–440. doi: 10.13374/j.issn2095–9389.2019.09.05.005

LIANG Shu, PENG Kaixiang. Distributed gradient-based consensus optimization algorithm and convergence analysis[J]. *Chinese Journal of Engineering*, 2020, 42(4): 434–440. doi: 10.13374/j.issn2095–9389.2019.09.05.005

在线阅读 View online: <https://doi.org/10.13374/j.issn2095–9389.2019.09.05.005>

您可能感兴趣的其他文章

Articles you may be interested in

纯电动车用18650电池的一致性研究

Consistency study on 18650 cells used in electric vehicles

工程科学学报. 2017, 39(1): 107 <https://doi.org/10.13374/j.issn2095–9389.2017.01.014>

确定性多变量自校正控制的稳定性、收敛性和鲁棒性

Stability, convergence, and robustness of deterministic multivariable self-tuning control

工程科学学报. 2019, 41(9): 1215 <https://doi.org/10.13374/j.issn2095–9389.2019.09.014>

混沌人工鱼群的鲁棒保性能控制权值矩阵优化方法

A weighting matrix optimization method for robust guaranteed cost control based on chaos artificial fish swarm algorithm

工程科学学报. 2018, 40(4): 500 <https://doi.org/10.13374/j.issn2095–9389.2018.04.014>

基于UWB的地下定位算法和拓扑优化

An underground localization algorithm and topology optimization based on ultra-wideband

工程科学学报. 2018, 40(6): 743 <https://doi.org/10.13374/j.issn2095–9389.2018.06.013>

基于自适应搜索的免疫粒子群算法

Immune particle swarm optimization algorithm based on the adaptive search strategy

工程科学学报. 2017, 39(1): 125 <https://doi.org/10.13374/j.issn2095–9389.2017.01.016>

分布式一致性最优化的梯度算法与收敛分析

梁 舒[✉], 彭开香

北京科技大学自动化学院工业过程知识自动化教育部重点实验室, 北京 100083

✉通信作者, E-mail: sliang@ustb.edu.cn

摘 要 研究了多智能体网络中受集合约束的一致性最优化问题, 提出了基于原始-对偶梯度的定步长分布式算法。算法中包括步长在内的参数会影响收敛性, 需要先进行收敛分析, 再根据收敛条件设置合适的参数。本文首先针对一般的定步长迭代格式, 提出一种基于李雅普诺夫函数的收敛分析范式, 它类似于一般微分方程关于李雅普诺夫稳定的分析方法。然后, 针对所考虑的分布式梯度算法, 构造了合适的李雅普诺夫函数, 并根据收敛条件得到了算法参数设定范围, 避免了繁冗复杂的分析论证。本文提出的理论与方法也为其他类型的分布式算法提供了一个框架性、系统性的论证方法。

关键词 分布式优化; 原始-对偶; 收敛分析; 李雅普诺夫函数; 一致性

分类号 TP27

Distributed gradient-based consensus optimization algorithm and convergence analysis

LIANG Shu[✉], PENG Kaixiang

Key Laboratory of Knowledge Automation for Industrial Processes of Ministry of Education, School of Automation, University of Science and Technology Beijing, Beijing 100083, China

✉ Corresponding author, E-mail: sliang@ustb.edu.cn

ABSTRACT A distributed optimization problem is cooperatively solved by a network of agents, which have significant applications in many science and engineering fields, such as metallurgical engineering. For complex industrial processes with multiple-level characteristics, varying working conditions, and long processes, numerous optimization decision-making micro and macro control problems, such as product quality control, production planning, scheduling, and energy comprehensive deployment, are encountered. The theory and method of distributed optimization are keys to promoting the strategic decision-making of the integration of industrialization and new-generation industrial revolution. Their development enhances the ability to deal with large-scale and complex problems of big data, which have important practical value and economic benefits. In this study, consensus optimization with set constraints in multi-agent networks was explored. A distributed algorithm with a fixed step size was proposed on the basis of a primal-dual gradient scheme. Parameters such as step size affect the convergence of the algorithm. As such, convergence should be analyzed first, and appropriate parameters should be subsequently set in accordance with convergence conditions. Existing works have constructed different Lyapunov functions by exploiting the specific iteration scheme of this algorithm and analyzing convergence. Conversely, a convergence analysis paradigm based on a Lyapunov function was proposed in this study for general fixed step size iteration schemes, which were similar to the analysis method of Lyapunov convergence for general differential equations. A suitable Lyapunov function was constructed for the distributed gradient algorithm, and a parameter setting range was obtained in accordance with the convergence conditions. The proposed method avoids the tedious and complicated analysis of algorithm convergence and parameter assignment. The theory and method presented in this study also provide a framework and systematic demonstration method for other types of distributed algorithms and may be regarded as future directions of distributed optimization.

收稿日期: 2019-09-05

基金项目: 国家自然科学基金资助项目(61903027, 61873024)

KEY WORDS distributed optimization; primal-dual; convergence analysis; Lyapunov function; consensus

分布式优化是多智能体系统控制、网络通信和数学规划的赛博空间(Cyberspace)科学,在众多科学与工程中具有广阔的发展前景^[1-2].以钢铁行业自动化为例,近年来我国钢铁生产企业普遍建立并实施了企业资源计划、生产执行系统、生产过程系统等多层次的集成自动化系统.对于具有多层次、变工况、长流程等特点的复杂工业过程,其产品质量管控、生产计划与调度、能源综合调配等微观与宏观调控方面存在大量的优化决策问题^[3].分布式优化理论与方法是促进两化融合战略决策和新一代工业革命的关键使能技术,其发展将增强人们对付大数据、大规模问题和复杂问题的能力,具有重要的实际应用意义并蕴藏着极大的经济效益.

分布式优化的一类抽象问题类型是一致性最优化,要求所有个体的决策变量最终实现一致性,并且一致点是一个凸优化问题的最优解. Nedic 等^[4-6]对该问题进行了较深入地研究,主要针对非光滑的目标函数,采用分布式次梯度的方法进行求解.其中,为了确保算法的收敛性,需要采用逐渐衰减并趋于零的步长. Shi 等^[7]针对无约束的光滑最优一致性,提出一种定步长并能精确收敛到最优解的分布式算法.该算法主要的思想是对所有个体的梯度之和进行跟踪,并利用不精确梯度理论对算法的收敛性进行分析.基于这种方法, Nedic 等^[8]改进了这种算法,并利用小增益的技术给出了算法的指数收敛证明.需要指出,这类算法仅处理无约束的分布式一致性最优化问题,目前还不能推广到有约束的问题. Liu 等^[9]和 Lei 等^[10]利用经典约束优化中的原始-对偶梯度方法,给出了分布式原始-对偶梯度算法.这类算法也采用了固定的步长,能够精确收敛到最优解,而且可以处理带有约束的问题. Liang 等^[11]针对无约束情形下的分布式原始-对偶梯度算法,基于变分分析中的度量次正则性,给出算法指数收敛的判据,降低了对目标函数强凸性的要求.此外,值得指出,有不少学者从连续时间微分方程的角度进行了分布式优化算法设计,如文献^[12-16].总之,关于分布式一致性最优化问题已有不少研究,而近年来学者们更加关注分布式定步长算法,以及对它们收敛性质的分析论证.

本文考虑带有约束的分布式一致性最优化问

题,给出定步长分布式原始-对偶梯度算法,并对算法的收敛性进行分析论证.最主要的贡献在于提出一种基于李雅普诺夫函数的收敛分析范式:针对一般的定步长迭代格式,给出基于李雅普诺夫函数的收敛条件,类似于一般微分方程关于李雅普诺夫稳定的分析方法.这种方法带来的好处在于将繁冗复杂的收敛性分析工作简化为构造李雅普诺夫函数并验证收敛条件.在此基础上,本文将该方法应用到分布式原始-对偶梯度算法中,给出算法参数应满足的收敛条件.与已有文献中的方法相比,如文献^[9]和^[10],虽然它们从不同的角度利用算法具体结构设计构造了李雅普诺夫函数并辅助收敛分析,但本文提出的方法具有框架性和系统性,对其他类型分布式优化算法的收敛分析也具有参考价值.

下面介绍本文的记号. \mathbb{R} 和 \mathbb{R}^n 分别代表实数空间和 n 维欧氏空间.记 $\langle x, y \rangle$ 为向量 x 和 y 的内积, $\| \cdot \|_2$ 和 $\| \cdot \|_\infty$ 分别代表2-范数和无穷范数.对于 N 个列向量 x_1, \dots, x_N ,使用 $\text{col}(x_1, \dots, x_N)$ 代表其堆栈而成的列向量.用 $\mathbf{0}_n$ 、 $\mathbf{1}_n$ 分别表示分量全部为0和分量全部为1的列向量,用 I_n 表示 $\mathbb{R}^{n \times n}$ 中的单位阵.对于矩阵 $A = [a_{ij}]$, a_{ij} 代表该矩阵在第 i 行第 j 列的元素.符号 \otimes 表示矩阵的Kronecker乘积.

1 预备知识

本节的主要目的是罗列后续论述所需的预备知识,包括凸集与凸函数,以及图论.

1.1 凸集与凸函数

本段介绍凸集及其投影的相关定义和性质,更详细的论述可参考专著^[17].设 Ω 是 n 维欧氏空间 \mathbb{R}^n 中的集合.若对任意 $x, y \in \Omega$ 和 $0 < \tau < 1$,均有 $\tau x + (1 - \tau)y \in \Omega$,则称 Ω 为凸集.若一个集合中任意点列的极限点仍然属于该集合,则称该集合是闭集.定义 $P_\Omega(z) \triangleq \text{argmin}_{x \in \Omega} \|z - x\|_2$ 为点 $z \in \mathbb{R}^n$ 到闭凸集合 Ω 的欧氏距离投影.投影算子具有一个基本性质: $\langle z - P_\Omega(z), P_\Omega(z) - x \rangle \geq 0, \forall z \in \mathbb{R}^n, \forall x \in \Omega$.

本段介绍凸函数的基本概念和性质,更多结论可参考文献^[18].设 $\Omega \subset \mathbb{R}^n$ 为凸集, $f: \Omega \rightarrow \mathbb{R}$ 为一个实函数.如果对任意 $x, y \in \Omega$ 和 $0 < \tau < 1$,均有 $f(\tau x + (1 - \tau)y) \leq \tau f(x) + (1 - \tau)f(y)$,则称 f 为凸函数.当 f 可微时,其梯度记为 $\nabla f(x)$.可微凸函数具有一个基本性质:

$$0 \leq f(y) - f(x) - \langle y - x, \nabla f(x) \rangle, \forall x, y \in \Omega$$

此外, 若映射 $\nabla f(x)$ 关于 x 是 κ -Lipschitz 连续的, 其中 $\kappa > 0$ 为某个常数, 则不论 f 是否是凸函数, 都有下面不等式成立:

$$f(y) - f(x) - \langle y - x, \nabla f(x) \rangle \leq \frac{\kappa}{2} \|y - x\|_2^2, \forall x, y \in \Omega$$

1.2 图论

网络节点之间的信息分享关系可以通过图论中的基本概念进行描述. 一个图 \mathcal{G} 由节点集合 \mathcal{V} 和边集 \mathcal{E} 构成, 记为 $\mathcal{G} = (\mathcal{V}, \mathcal{E})$. 对于具有 N 个节点的网络, 通常记 $\mathcal{V} = \{1, \dots, N\}$. 边集 $\mathcal{E} \subset \mathcal{V} \times \mathcal{V}$ 刻画了节点之间的信息分享关系: 若无序节点对 $(i, j) \in \mathcal{E}$, 那么节点 i 和节点 j 可以相互通信. 如果对于任意 $i, j \in \mathcal{V}$, 都可以从节点 i 出发通过若干条边连接到节点 j , 则称 \mathcal{G} 为连通图. 另外, 可以通过一些代数数量对图进行描述. 定义图 \mathcal{G} 的加权邻接矩阵 $A = [a_{ij}]$: 当 $(i, j) \in \mathcal{E}$ 时, $a_{ij} = a_{ji} > 0$, 否则 $a_{ij} = 0$. 对于任意 $i \in \mathcal{V}$, 节点 i 的度定义为 $d_i \triangleq \sum_{j=1}^N a_{ij}$. 进一步, 图的度矩阵定义为 $D \triangleq \text{diag}\{d_1, \dots, d_N\}$, 图的 Laplace 矩阵定义为 $L \triangleq D - A$. 矩阵 L 是对称半正定矩阵, 且 $L\mathbf{1}_N = \mathbf{0}_n$. 而且, 若图 \mathcal{G} 是连通的, 则 L 仅有一个零特征值, 其他特征值都为正数. 有关图论和代数图论的详细内容可以参考文献 [19].

2 优化问题与分布式算法

本节首先描述分布式一致性最优化问题, 然后给出求解该问题的分布式算法.

2.1 问题描述

考虑由 N 个个体组成的多智能体网络, 其信息共享关系图为 $\mathcal{G} = (\mathcal{V}, \mathcal{E})$. 对于 $i \in \mathcal{V}$, 个体 i 可以控制本地的决策变量 $x_i \in \Omega_i$, 其中, $\Omega_i \subset \mathbb{R}^n$ 是本地可行集合. 多智能体一致性控制问题, 是指设计分布式的控制律, 使得所有决策变量可以从初态收敛到某个相同的末态, 即渐近满足 $x_1 = \dots = x_N$. 这里的分布式, 是指仅允许从本地数据和网络邻居个体中获得信息来更新决策变量. 进一步, 考虑每个个体都有一个关于决策变量的目标函数 (成本函数或效用函数), 记为 $f_i(x_i)$. 分布式一致性最优化问题, 是希望设计分布式算法, 使得多智能体网络不仅能够实现一致性, 还能实现总体目标函数的最优性.

结合上述多智能体网络背景, 分布式一致性最优化问题的数学描述为

$$\min_{x \in \Omega} f(x), \text{ s.t. } x_1 = \dots = x_N \in \Omega_0 \quad (1)$$

其中, $x \triangleq \text{col}(x_1, \dots, x_N)$, $f(x) \triangleq f_1(x_1) + \dots + f_N(x_N)$, $\Omega \triangleq \prod_{i=1}^N \Omega_i$, $\Omega_0 \triangleq \bigcap_{i=1}^N \Omega_i$.

在求解问题 (1) 的过程中, 假定每个个体 $i \in \mathcal{V}$ 的本地目标函数不能被全局共享或者被其他个体获取, 而个体 i 能够根据本地数据获取目标函数在给定点的梯度值, 但无法获取其他个体目标函数相关的任何信息.

下面给出问题 (1) 的基本假设条件.

假设 1: 对于每个 $i \in \mathcal{V}$, $f_i(x_i)$ 是可微凸函数, 且 $\nabla f_i(x_i)$ 是 κ_f -Lipschitz 连续的, 其中 $\kappa_f > 0$ 为某个常数. Ω_i 是非空的闭凸集合. 并且, 优化问题 (1) 至少存在一个有限的最优解.

假设 2: 图 \mathcal{G} 是一个连通图.

上述条件是最基本的, 也常见于其他文献中. 在假设 1 中, f_i 与 Ω_i 的凸性保证问题 (1) 是一个凸优化问题. f_i 的可微性使得个体能够利用本地梯度信息进行寻优. 最优解集非空则确保了问题的适定性. 此外, 假设 2 中的连通性使多智能体网络实现整体一致性和最优性成为可能.

注 1: 与文献 [7]、[8] 和 [11] 相比, 本文的分布式优化的问题模型带有集合约束, 所以具有更广的应用范围. 文献 [4] 考虑了集合约束, 但仅对相同的约束情形进行了分析, 即 $\Omega_1 = \dots = \Omega_N$. 而本文不需要该限制条件.

2.2 分布式算法

为了能够借助凸优化理论与算法对问题 (1) 进行求解, 先将一致性等式约束 $x_1 = \dots = x_N$ 等价地转化为标准等式约束形式 $h(x) = \mathbf{0}$. 事实上, 可以选择 $h(x) = L \otimes I_n x$, 其中 L 是图 \mathcal{G} 的 Laplace 矩阵. 这是因为, 在假设 2 的条件下, $L \otimes I_n x = \mathbf{0}_{nN}$ 当且仅当 $x_1 = \dots = x_N$. 于是, 问题 (1) 的等价描述为

$$\min_{x \in \Omega} f(x), \text{ s.t. } L \otimes I_n x = \mathbf{0}_{nN} \quad (2)$$

问题 (2) 是一个标准的约束优化问题, 可以定义一个 Lagrange 函数

$$\mathcal{L}(x, \lambda) \triangleq f(x) + \langle \lambda, L \otimes I_n x \rangle + \frac{\beta}{2} \langle x, L \otimes I_n x \rangle \quad (3)$$

其中, $\lambda \triangleq \text{col}(\lambda_1, \dots, \lambda_N) \in \mathbb{R}^{nN}$ 为 Lagrange 乘子, $\beta > 0$ 为一个常数. 通常也把 x 和 λ 分别称为原始变量和对偶变量. 在假设 1 的条件下, 最优化问题 (2) 的一阶原始-对偶最优条件为

$$\begin{aligned} \mathbf{0}_{nN} &= P_{\Omega}(x - \alpha \nabla_x \mathcal{L}(x, \lambda)) - x = \\ P_{\Omega}(x - \alpha (\nabla f(x) + \beta L \otimes I_n x + L \otimes I_n \lambda)) - x & \quad (4) \\ \mathbf{0}_{nN} &= \nabla_{\lambda} \mathcal{L}(x, \lambda) = L \otimes I_n x \end{aligned}$$

其中, $\alpha > 0$ 为任意常数.

根据最优条件(4), 可以设计一个基于原始-对偶梯度的迭代算法:

$$\begin{aligned} \mathbf{x}_i[k+1] &= P_{\Omega_i}(\mathbf{x}_i[k] - \\ &\alpha \left(\nabla f_i(\mathbf{x}_i[k]) + \beta \sum_{j=1}^N a_{ij}(\mathbf{x}_i[k] - \mathbf{x}_j[k]) + \right. \\ &\left. \sum_{j=1}^N a_{ij}(\lambda_i[k] - \lambda_j[k]) \right)) \\ \lambda_i[k+1] &= \lambda_i[k] + \alpha \sum_{j=1}^N a_{ij}(\mathbf{x}_i[k] - \mathbf{x}_j[k]) \end{aligned} \quad (5)$$

其中, 常数 α 和 β 待定, 它们将在收敛分析中根据收敛条件来确定. 算法(5)是一个分布式算法. 这是因为, 个体 i 在更新 \mathbf{x}_i 和 λ_i 的过程中, 仅利用了本地数据 $\nabla f_i(\mathbf{x}_i)$ 、 Ω_i 、 \mathbf{x}_i 和 λ_i , 以及邻居个体的信息 \mathbf{x}_j 和 λ_j .

将所有个体决策变量和对偶变量罗列起来, 写成更紧凑的形式:

$$\begin{aligned} \mathbf{x}[k+1] &= P_{\Omega}(\mathbf{x}[k] - \alpha \nabla_{\mathbf{x}} \mathcal{L}(\mathbf{x}[k], \boldsymbol{\lambda}[k])) \\ \boldsymbol{\lambda}[k+1] &= \boldsymbol{\lambda}[k] + \alpha \nabla_{\boldsymbol{\lambda}} \mathcal{L}(\mathbf{x}[k], \boldsymbol{\lambda}[k]) \end{aligned} \quad (6)$$

由式(6)可见, 算法的设计上采用了求解 \mathcal{L} 鞍点的梯度下降-梯度上升的方法. 其不动点满足一阶原始-对偶最优条件(4), 从而确保了算法的正确性.

注 2: 文献 [9-10] 提出了类似的分布式原始-对偶梯度算法. 相比之下, 本文考虑的算法具有两个可调参数: α 和 β . 因此, 该算法更具灵活性, 并且推广了文献中的算法.

3 收敛性分析

本节首先对一般的定步长迭代格式, 提出基于李雅普诺夫函数的一种分析范式. 在此基础上, 对本文的分布式算法进行分析, 给出算法参数的选择范围.

3.1 基于李雅普诺夫函数的一种分析范式

考虑一般的迭代格式

$$\mathbf{z}[k+1] = \mathbf{z}[k] - \alpha \mathbf{F}(\mathbf{z}[k]), \mathbf{z}[0] = \mathbf{z}_0 \in \mathcal{Z} \subset \mathbb{R}^m \quad (7)$$

其中 $\mathbf{F}: \mathbb{R}^m \rightarrow \mathbb{R}^m$ 为连续映射, $\alpha > 0$ 为常步长. 集合 \mathcal{Z} 为迭代格式(7)的不变集, 即 $\mathbf{z}[k] \in \mathcal{Z}, \forall k \geq 0$. 令 $\mathcal{Z}^* \triangleq \{\mathbf{z} \in \mathcal{Z} | \mathbf{F}(\mathbf{z}) = \mathbf{0}_m\}$ 为平衡点集合, 这里假定 \mathcal{Z}^* 不为空集.

为了分析论证迭代(7)的收敛性, 本文提出一种基于李雅普诺夫函数的分析范式.

定理 1: 如果存在一个李雅普诺夫函数 $V: \mathbb{R}^m \times \mathbb{R}^m \rightarrow \mathbb{R}$ 满足:

1) 对于任意的 $\mathbf{z}^* \in \mathcal{Z}^*$, $V(\mathbf{z}, \mathbf{z}^*)$ 是正定的, 即

$V(\mathbf{z}, \mathbf{z}^*) \geq 0$, 且 $V(\mathbf{z}, \mathbf{z}^*) = 0 \Leftrightarrow \mathbf{z} = \mathbf{z}^*$;

2) 对于任意的 $\mathbf{z}^* \in \mathcal{Z}^*$, $V(\mathbf{z}, \mathbf{z}^*)$ 关于 \mathbf{z} 是水平强制的, 即对于任意的正数 $\gamma > 0$, V 的水平集合 $\text{lev}_{\leq \gamma} V \triangleq \{\mathbf{z} \in \mathcal{Z} | V(\mathbf{z}, \mathbf{z}^*) \leq \gamma\}$ 有界;

3) 对于任意的 $\mathbf{z}^* \in \mathcal{Z}^*$, $V(\mathbf{z}, \mathbf{z}^*)$ 关于 \mathbf{z} 可微, 且映射 $\nabla_{\mathbf{z}} V(\mathbf{z}, \mathbf{z}^*)$ 在 $\mathbf{z} \in \mathcal{Z}$ 上是 κ_V -Lipschitz连续的, 其中 $\kappa_V > 0$ 是一个常数;

4) 存在某个常数 $\delta > 0$, 使得对于任意的 $\mathbf{z}^* \in \mathcal{Z}^*$, 都有 $\langle \mathbf{F}(\mathbf{z}), \nabla_{\mathbf{z}} V(\mathbf{z}, \mathbf{z}^*) \rangle \geq \delta \|\mathbf{F}(\mathbf{z})\|_2^2, \forall \mathbf{z} \in \mathcal{Z}$, 其中 $\mathbf{F}(\mathbf{z})$ 是(7)中的映射;

5) 迭代格式(7)中, 步长满足 $0 < \alpha < \frac{2\delta}{\kappa_V}$.

那么, 由算法(7)产生的序列 $\{\mathbf{z}[k]\}$ 收敛到某一个平衡点, 即存在某个 $\tilde{\mathbf{z}}^* \in \mathcal{Z}^*$, 满足

$$\lim_{k \rightarrow \infty} \mathbf{z}[k] = \tilde{\mathbf{z}}^* \quad (8)$$

证明: 任取 $\mathbf{z}^* \in \mathcal{Z}^*$. 因为 $\nabla_{\mathbf{z}} V(\mathbf{z}, \mathbf{z}^*)$ 是 κ_V -Lipschitz连续的, 有

$$\begin{aligned} V(\mathbf{z}[k+1], \mathbf{z}^*) - V(\mathbf{z}[k], \mathbf{z}^*) &\leq \\ \langle \mathbf{z}[k+1] - \mathbf{z}[k], \nabla_{\mathbf{z}} V(\mathbf{z}[k], \mathbf{z}^*) \rangle + \\ \frac{\kappa_V}{2} \|\mathbf{z}[k+1] - \mathbf{z}[k]\|_2^2 &= -\alpha \langle \mathbf{F}(\mathbf{z}[k]), \\ \nabla_{\mathbf{z}} V(\mathbf{z}[k], \mathbf{z}^*) \rangle + \alpha^2 \frac{\kappa_V}{2} \|\mathbf{F}(\mathbf{z}[k])\|_2^2 \end{aligned}$$

又因为 $\langle \mathbf{F}(\mathbf{z}), \nabla_{\mathbf{z}} V(\mathbf{z}, \mathbf{z}^*) \rangle \geq \delta \|\mathbf{F}(\mathbf{z})\|_2^2$, 所以

$$V(\mathbf{z}[k+1], \mathbf{z}^*) - V(\mathbf{z}[k], \mathbf{z}^*) \leq -\frac{\alpha(2\delta - \alpha\kappa_V)}{2} \|\mathbf{F}(\mathbf{z}[k])\|_2^2 \leq 0$$

由此可见, $V(\mathbf{z}[k], \mathbf{z}^*)$ 是随着 k 的增大而单调递减的, 从而 $V(\mathbf{z}[k], \mathbf{z}^*) \leq V(\mathbf{z}[0], \mathbf{z}^*)$ 有界. 由于 $V(\mathbf{z}, \mathbf{z}^*)$ 是水平强制的, 因此序列 $\{\mathbf{z}[k]\}$ 有界. 不妨设 $\tilde{\mathbf{z}}^* \in \mathcal{Z}$ 是当 $k \rightarrow \infty$ 时序列 $\{\mathbf{z}[k]\}$ 的一个聚点, 即存在子列 $k_l \rightarrow \infty$, 使得 $\lim_{l \rightarrow \infty} \mathbf{z}[k_l] = \tilde{\mathbf{z}}^*$. 注意到

$$\begin{aligned} \frac{\alpha(2\delta - \alpha\kappa_V)}{2} \sum_{k=0}^{\infty} \|\mathbf{F}(\mathbf{z}[k])\|_2^2 &\leq \limsup_{K \rightarrow \infty} \\ \sum_{k=0}^K V(\mathbf{z}[k], \mathbf{z}^*) - V(\mathbf{z}[k+1], \mathbf{z}^*) &\leq V(\mathbf{z}[0], \mathbf{z}^*) < +\infty \end{aligned}$$

所以, $\lim_{k \rightarrow \infty} \mathbf{F}(\mathbf{z}[k]) = \mathbf{0}_m$. 又因为 \mathbf{F} 是连续映射, 所以 $\mathbf{F}(\tilde{\mathbf{z}}^*) = \mathbf{0}_m$. 这表明 $\tilde{\mathbf{z}}^* \in \mathcal{Z}^*$.

重新选取李雅普诺夫函数 $V(\mathbf{z}, \tilde{\mathbf{z}}^*)$, 按照类似的证明过程, 可以得出 $\{V(\mathbf{z}[k], \tilde{\mathbf{z}}^*)\}$ 也是单调递减的序列. 不仅如此, 由于 $V(\tilde{\mathbf{z}}^*, \tilde{\mathbf{z}}^*) = 0$, 序列 $\{V(\mathbf{z}[k], \tilde{\mathbf{z}}^*)\}$ 有一个收敛到零的子列. 从而可以得出序列 $\{V(\mathbf{z}[k], \tilde{\mathbf{z}}^*)\}$ 单调递减并趋于零. 再根据 $V(\mathbf{z}, \tilde{\mathbf{z}}^*)$ 的正定性, 就得到 $\lim_{k \rightarrow \infty} \mathbf{z}[k] = \tilde{\mathbf{z}}^*$. 证毕!

3.2 分布式一致性最优优化梯度算法的收敛分析

考虑分布式一致性最优优化梯度算法(6). 为了得收敛条件, 采用上小节提出的基于李雅普诺夫函数的分析方法.

令 $z \triangleq \text{col}(x, \lambda)$. 定义 Z^* 为原始-对偶最优解集合, 即所有满足等式 (4) 的点 $z^* = \text{col}(x^*, \lambda^*)$ 的集合. 在迭代格式 (6) 中, $\{x[k]\}$ 始终保持在集合 Ω 中, 因此 $Z \triangleq \Omega \times \mathbb{R}^{nN}$ 是一个不变集. 可以将 (6) 重新写成类似 (7) 的标准迭代格式

$$z[k+1] = z[k] - \alpha F(z[k]), z[0] = z_0 \in Z \quad (9)$$

其中,

$$F(z) \triangleq \frac{z - P_Z(z - \alpha G(z))}{\alpha}, G(z) \triangleq \begin{bmatrix} \nabla_x \mathcal{L}(x, \lambda) \\ -\nabla_\lambda \mathcal{L}(x, \lambda) \end{bmatrix} \quad (10)$$

根据最优条件, 映射 $G(z)$ 满足

$$\langle z - z^*, G(z^*) \rangle \geq 0, \forall z \in Z, \forall z^* \in Z^* \quad (11)$$

接下来, 构造李雅普诺夫函数

$$V(z, z^*) = \alpha(\mathcal{L}(x, \lambda) - \mathcal{L}(x^*, \lambda^*)) - \langle x - x^*, \nabla_x \mathcal{L}(x^*, \lambda^*) \rangle + \frac{1}{2} \|z - z^*\|_2^2 \quad (12)$$

由于 $\mathcal{L}(x, \lambda^*)$ 是关于 x 的凸函数, 所以 $\mathcal{L}(x, \lambda^*) - \mathcal{L}(x^*, \lambda^*) - \langle x - x^*, \nabla_x \mathcal{L}(x^*, \lambda^*) \rangle \geq 0$. 于是,

$$V(z, z^*) \geq \alpha(\mathcal{L}(x, \lambda) - \mathcal{L}(x, \lambda^*)) + \frac{1}{2} \|z - z^*\|_2^2 \geq \frac{1 - \alpha \|L\|_\infty}{2} \|z - z^*\|_2^2 \quad (13)$$

而且, $V(z, z^*)$ 是关于 z 的可微函数, 可以通过计算得到

$$\begin{aligned} \nabla_z V(z, z^*) &= \alpha \left(\begin{bmatrix} \nabla_x \mathcal{L}(x, \lambda) \\ \nabla_\lambda \mathcal{L}(x, \lambda) \end{bmatrix} - \begin{bmatrix} \nabla_x \mathcal{L}(x^*, \lambda^*) \\ -\nabla_\lambda \mathcal{L}(x^*, \lambda^*) \end{bmatrix} \right) + z - z^* \\ &= \alpha \begin{bmatrix} \nabla f(x) - \nabla f(x^*) + \beta L \otimes I_n x + L \otimes I_n (\lambda - \lambda^*) \\ L \otimes I_n x \end{bmatrix} + z - z^* \end{aligned} \quad (14)$$

容易证明, $\nabla_z V(z, z^*)$ 是关于 z 的 κ_V -Lipschitz 连续映射, 其中

$$\kappa_V = \alpha(\kappa_f + \beta \|L\|_\infty + \|L\|_\infty) + 1 \quad (15)$$

下面验证不等式 $\langle F(z), \nabla_z V(z, z^*) \rangle \geq \delta \|F(z)\|_2^2$, $\forall z \in Z$, 并确定常数 δ . 通过计算,

$$\begin{aligned} \langle F(z), \nabla_z V(z, z^*) \rangle &= \frac{1}{\alpha} \langle z - P_Z(z - \alpha G(z)), \\ &\alpha \left(\begin{bmatrix} \nabla_x \mathcal{L}(x, \lambda) \\ \nabla_\lambda \mathcal{L}(x, \lambda) \end{bmatrix} - \begin{bmatrix} \nabla_x \mathcal{L}(x^*, \lambda^*) \\ -\nabla_\lambda \mathcal{L}(x^*, \lambda^*) \end{bmatrix} \right) + z - z^* \rangle \\ &= \frac{1}{\alpha} \langle z - P_Z(z - \alpha G(z)), \alpha G(z) - \alpha G(z^*) + z - z^* \rangle \\ &\quad - 2\alpha \|\nabla_\lambda \mathcal{L}(x, \lambda)\|_2^2 \end{aligned} \quad (16)$$

根据投影算子的基本性质和不等式 (11), 容易证明

$$\begin{aligned} \frac{1}{\alpha} \langle z - P_Z(z - \alpha G(z)), \alpha G(z) - \alpha G(z^*) + z - z^* \rangle &\geq \\ \frac{1}{\alpha} \|z - P_Z(z - \alpha G(z))\|_2^2 + \langle z - z^*, G(z) - G(z^*) \rangle &\end{aligned} \quad (17)$$

并且

$$\langle z - z^*, G(z) - G(z^*) \rangle = \langle x - x^*, \nabla f(x) - \nabla f(x^*) \rangle + \beta \langle x, L \otimes I_n x \rangle \geq \beta \langle x, L \otimes I_n x \rangle \quad (18)$$

与此同时, 通过简单计算得到 $\|\nabla_\lambda \mathcal{L}(x, \lambda)\|_2^2 = \|L \otimes I_n x\|_2^2$, 且

$$\|L\|_\infty \langle x, L \otimes I_n x \rangle \geq \|L \otimes I_n x\|_2^2, \forall x \in \mathbb{R}^{nN} \quad (19)$$

结合式 (16) ~ (19), 得到

$$\begin{aligned} \langle F(z), \nabla_z V(z, z^*) \rangle &\geq \frac{1}{\alpha} \|z - P_Z(z - \alpha G(z))\|_2^2 + \\ \left(\frac{\beta}{\|L\|_\infty} - 2\alpha \right) \|L \otimes I_n x\|_2^2 &\end{aligned} \quad (20)$$

至此, 可以给出分布式算法 (5) 的收敛分析结果.

定理 2: 在假设 1 和假设 2 的条件下, 如果分布式算法 (5) 的步长 α 和常数 β 满足

$$0 < \alpha < \min \left\{ \frac{1}{\kappa_f + \beta \|L\|_\infty + \|L\|_\infty}, \frac{\beta}{2\|L\|_\infty} \right\}, \beta > 0 \quad (21)$$

那么算法产生的迭代序列收敛到某个原始-对偶最优解.

证明: 因为已经给出了定理 1 所述的基于李雅普诺夫函数的分析范式, 所以这里仅需考虑式 (12) 构造的 $V(z, z^*)$, 并验证其满足各项条件.

首先根据 (13), 若 $0 < \alpha < 1/\|L\|_\infty$, 则 $V(z, z^*)$ 是正定的, 也是水平强制的, 即满足范式条件中的 1) 和 2). 然后, 由 (15) 可知, $V(z, z^*)$ 满足条件 3). 再次, 根据式 (20), 如果 $\beta \geq 2\alpha\|L\|_\infty$, 则

$$\langle F(z), \nabla_z V(z, z^*) \rangle \geq \alpha \|F(z)\|_2^2 \quad (22)$$

即 $V(z, z^*)$ 满足条件 4), 且 $\delta = \alpha$. 将范式条件 5) 及其他关于 α 和 β 的条件列在一起, 得到

$$\begin{cases} 0 < \alpha < 1/\|L\|_\infty, \beta \geq 2\alpha\|L\|_\infty \\ \alpha < \frac{2\delta}{\kappa_V} = \frac{2\alpha}{\alpha(\kappa_f + \beta\|L\|_\infty + \|L\|_\infty) + 1} \end{cases}$$

整理后即可得到式 (21) 中的参数条件.

证毕!

4 数值实验

本节给出一个数值算例来验证算法的有效性.

考虑优化问题 (2), 其中 $N = 5$, 决策变量 $x_i \in \mathbb{R}$, $\Omega_i = [i - 10, i - 2]$, $i \in \{1, 2, 3, 4, 5\}$. 每个个体的本地目标函数为

$$f_i(x) = \begin{cases} \frac{i}{6}(-x + i - 5)^2, & x < i - 5 \\ 0, & i - 5 \leq x \leq i + 5 \\ \frac{i}{6}(x - i - 5)^2, & x > i + 5 \end{cases}$$

个体之间的信息分享关系如图 1 所示.

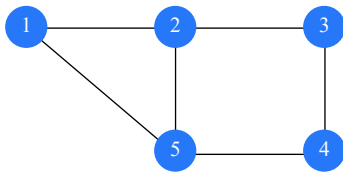


图 1 信息分享关系图
Fig.1 Information sharing graph

容易验证 $\Omega_0 = \bigcap_{i=1}^N \Omega_i = [-5, -1]$, 而问题的最优解为

$$x_1^* = \dots = x_5^* = -1 \quad (23)$$

通过分析和计算得

$$\kappa_f = \frac{5}{3}, \|\mathbf{L}\|_\infty = 6 \quad (24)$$

因此, 根据式 (21), 选择 $\beta = 0.9$, 得到参数 α 的选择范围是 $0 < \alpha < 0.075$. 在数值实验中, 选择 $\alpha = 0.07$, 得到的实验结果如图 2~4 所示. 可以看出, 由于选择了合适的步长等参数, 算法仅迭代 300 次后就已经收敛到最优解. 这些实验结果验证了本文提出的分布式算法及其收敛分析方法的正确性和有效性.

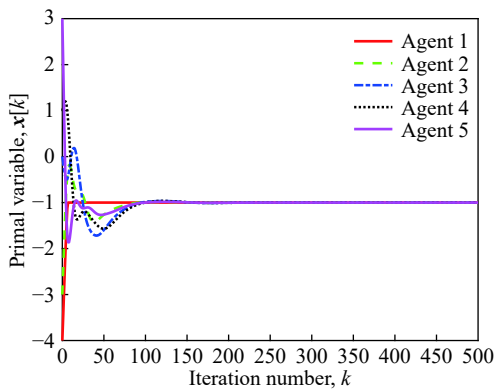


图 2 决策变量的更新轨迹

Fig.2 Trajectories of decision variables

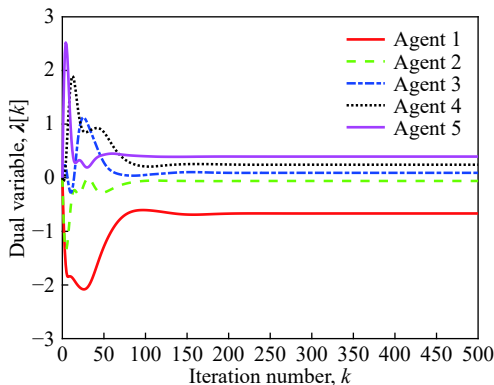


图 3 对偶变量的更新轨迹

Fig.3 Trajectories of dual variables

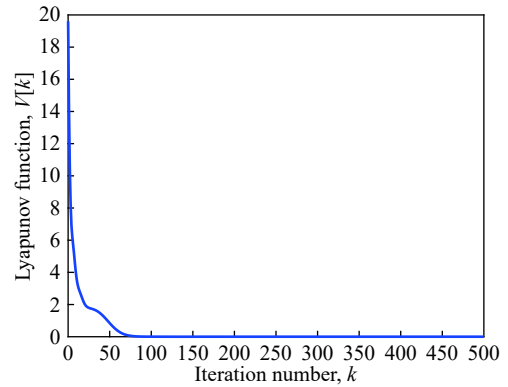


图 4 李雅普诺夫函数的更新轨迹

Fig.4 Trajectory of the Lyapunov function

5 结论

对分布式一致性最优化问题进行了研究, 设计了具有两个可调参数的分布式原始-对偶梯度算法, 并论证了算法的收敛性. 针对一般的定步长迭代格式, 提出一种基于李雅普诺夫函数的分析范式, 具有普适性和一般性, 在收敛分析方面具有框架性和系统性. 提出的方法运用在论文所考虑分布式梯度算法上, 得到了算法收敛条件, 确定出算法参数的选择范围, 证实了方法的有效性. 给出的数值实验结果也验证了该方法的有效性. 接下来的工作将包括对其他类型的分布式优化算法进行设计和分析, 如带有耦合等式或不等式约束的分布式资源分配问题等.

参 考 文 献

- [1] Yang T, Yi X L, Wu J F, et al. A survey of distributed optimization. *Ann Rev Control*, 2019, 47: 278
- [2] Yi P, Hong Y G. Distributed cooperative optimization and its applications. *Sci Sin Math*, 2016, 46(10): 1547 (衣鹏, 洪奕光. 分布式合作优化及其应用. 中国科学: 数学, 2016, 46(10): 1547)
- [3] Peng K X, Zhang C F, Ma L, et al. System-levels-based holographic fault diagnosis for complex industrial processes. *CIESC J*, 2019, 70(2): 590 (彭开香, 张传放, 马亮, 等. 面向系统层级的复杂工业过程全息故障诊断. 化工学报, 2019, 70(2): 590)
- [4] Nedic A, Ozdaglar A, Parrilo P A. Constrained consensus and optimization in multi-agent networks. *IEEE Trans Autom Control*, 2010, 55(4): 922
- [5] Nedic A, Olshevsky A. Distributed optimization over time-varying directed graphs. *IEEE Trans Autom Control*, 2015, 60(3): 601
- [6] Nedic A, Olshevsky A. Stochastic gradient-push for strongly convex functions on time-varying directed graphs. *IEEE Trans Autom Control*, 2016, 61(12): 3936

- [7] Shi W, Ling Q, Wu G, et al. Extra: An exact first-order algorithm for decentralized consensus optimization. *SIAM J Optim*, 2015, 25(2): 944
- [8] Nedic A, Olshevsky A, Shi W. Achieving geometric convergence for distributed optimization over time-varying graphs. *SIAM J Optim*, 2017, 27(4): 2597
- [9] Liu Q S, Yang S F, Hong Y G. Constrained consensus algorithms with fixed step size for distributed convex optimization over multi-agent networks. *IEEE Trans Autom Control*, 2017, 62(8): 4259
- [10] Lei J L, Chen H F, Fang H T. Primal-dual algorithm for distributed constrained optimization. *Syst Control Lett*, 2016, 96: 110
- [11] Liang S, Wang L Y, Yin G. Exponential convergence of distributed primal-dual convex optimization algorithm without strong convexity. *Automatica*, 2019, 105: 298
- [12] Ghahserifard B, Cortes J. Distributed continuous-time convex optimization on weight-balanced digraphs. *IEEE Trans Autom Control*, 2014, 59(3): 781
- [13] Zeng X L, Yi P, Hong Y G. Distributed continuous-time algorithm for constrained convex optimizations *via* nonsmooth analysis approach. *IEEE Trans Autom Control*, 2017, 62(10): 5227
- [14] Yang S F, Liu Q S, Wang J. A multi-agent system with a proportional-integral protocol for distributed constrained optimization. *IEEE Trans Autom Control*, 2017, 62(7): 3461
- [15] Deng Z H, Liang S, Hong Y G. Distributed continuous-time algorithms for resource allocation problems over weight-balanced digraphs. *IEEE Trans Cybern*, 2018, 48(11): 3116
- [16] Liang S, Zeng X L, Hong Y G. Distributed nonsmooth optimization with coupled inequality constraints *via* modified Lagrangian function. *IEEE Trans Autom Control*, 2018, 63(6): 1753
- [17] Hiriart-Urruty J B, Lemaréchal C. *Convex Analysis and Minimization Algorithms I: Fundamentals*. Springer Science & Business Media, 2013
- [18] Nesterov Y. *Lectures on Convex Optimization*. Springer, 2018
- [19] Godsil C, Royle G F. *Algebraic Graph Theory*. Springer Science & Business Media, 2013