

控制与决策

Control and Decision

基于预设时间收敛的分布式优化算法

杨涛, 常怡然, 张坤朋, 徐磊

引用本文:

杨涛,常怡然,张坤朋,徐磊. 基于预设时间收敛的分布式优化算法[J]. *控制与决策*, 2023, 38(8): 2364–2374.

在线阅读 View online: <https://doi.org/10.13195/j.kzyjc.2023.0391>

您可能感兴趣的其他文章

Articles you may be interested in

丢包和量化约束下的不确定系统分布式滚动时域估计

Distributed moving horizon estimation for stochastic uncertain system with packet dropouts and quantized measurements

控制与决策. 2021, 36(7): 1771–1778 <https://doi.org/10.13195/j.kzyjc.2019.1603>

一种自适应拟牛顿-状态转移混合智能优化算法及应用

A hybrid state transition optimization algorithm based on adaptive quasi-newton method and its application

控制与决策. 2021, 36(10): 2451–2458 <https://doi.org/10.13195/j.kzyjc.2020.0214>

多航天器系统分布式固定时间输出反馈姿态协同跟踪控制

Distributed fixed-time output feedback attitude coordination tracking control for multiple rigid spacecraft

控制与决策. 2021, 36(5): 1049–1058 <https://doi.org/10.13195/j.kzyjc.2019.0968>

孤岛微电网异构电池储能系统的分布式有限时间次级控制

Distributed finite-time secondary control for heterogeneous battery energy storage systems in an islanded microgrid

控制与决策. 2021, 36(8): 2034–2041 <https://doi.org/10.13195/j.kzyjc.2020.0012>

双重驱动的果蝇优化算法及其在PID控制器中的应用

Double drive fruit fly optimization algorithm and its application in PID controller

控制与决策. 2021, 36(9): 2225–2233 <https://doi.org/10.13195/j.kzyjc.2020.0046>

基于预设时间收敛的分布式优化算法

杨涛[†], 常怡然, 张坤朋, 徐磊

(东北大学 流程工业综合自动化国家重点实验室, 沈阳 110819)

摘要: 考虑一类分布式优化问题, 其目标是通过局部信息交互, 使得局部成本函数之和构成的全局成本函数最小. 针对该类问题, 通过引入时基发生器 (TBG), 提出两种基于预设时间收敛的分布式比例积分 (PI) 优化算法. 与现有的基于有限/固定时间收敛的分布式优化算法相比, 所提出算法的收敛时间不依赖于系统的初值和参数, 且可以任意预先设计. 此外, 在全局成本函数关于最优值点有限强凸, 局部成本函数为可微的凸函数, 且具有局部 Lipschitz 梯度的条件下, 通过 Lyapunov 理论证明了所提算法都能实现预设时间收敛. 最后, 通过数值仿真实验验证了所提出算法的有效性.

关键词: 分布式优化; 预设时间收敛; 比例积分算法; 有限强凸; 时基发生器

中图分类号: TP273

文献标志码: A

DOI: 10.13195/j.kzyjc.2023.0391

引用格式: 杨涛, 常怡然, 张坤朋, 等. 基于预设时间收敛的分布式优化算法[J]. 控制与决策, 2023, 38(8): 2364-2374.

Distributed optimization algorithms based on predefined-time convergence

YANG Tao[†], CHANG Yi-ran, ZHANG Kun-peng, XU Lei

(The State Key Laboratory of Synthetical Automation for Process Industries, Northeastern University, Shenyang 110819, China)

Abstract: This paper studies a class of distributed optimization problems, which aims to minimize the global cost function consisting of the sum of local cost functions through local information exchanges. For this class of problems, by introducing a time-based generator (TBG), the paper proposes two distributed proportional-integral (PI) optimization algorithms based on predefined-time convergence. Compared to existing distributed optimization algorithms based on finite/fixed time convergence, the convergence time of the proposed algorithms does not depend on initial values and parameters of the system and it can be arbitrarily predefined. Furthermore, the proposed algorithms can converge within a predefined time based on the Lyapunov theory under the conditions that the global cost function is restricted strongly convex with respect to the global optimal point along with local cost functions being convex, differentiable, and having local Lipschitz gradient. Finally, the effectiveness of these two algorithms is verified by numerical simulation.

Keywords: distributed optimization; predefined-time convergence; proportional-integral algorithm; restricted strongly convex; time-based generator

0 引言

计算机技术、大数据和无线通讯网络的兴起, 推动了多智能体系统和分布式优化的快速发展. 多智能体系统是由多个智能体通过网络关联形成, 其中每个智能体具备一定的感知、通信、计算和执行能力. 分布式优化算法的目标是通过多智能体系统邻居智能体间的局部信息交互, 使得所有智能体的局部成本函数之和所构成的全局成本函数达到最小. 该类算法的研究最早可追溯到文献[1-2]. 与集中式优

化算法相比, 分布式优化算法由于在降低通讯负载、保护数据隐私和提高系统鲁棒性等方面具有显著优势, 被广泛应用于电力系统^[3]、机器学习^[4]和传感器网络^[5]等领域^[6-10].

早期学者提出分布式(次)梯度下降 (distributed (sub)gradient descent, DGD) 算法^[11], 该算法由一致性项和梯度下降项组成. 由于该算法采用了衰减步长, 导致收敛速度较慢, 为了加快算法的收敛速度, 学者们对 DGD 算法进行了改进. 文献[12]采用固定步长

收稿日期: 2023-03-31; 录用日期: 2023-06-14.

基金项目: 国家自然科学基金重点项目(62133003); 国家自然科学基金重大项目(61991403, 61991400).

责任编辑: 杨光红.

[†]通讯作者. E-mail: yangtao@mail.neu.edu.cn.

梯度下降,虽然加快了收敛速度,却牺牲了收敛精度,只能收敛到全局最优值点的邻域内.为此,学者们通过增加校正项^[13]或采用动态梯度跟踪方法^[14-15]校正了固定步长导致的收敛误差.另外,文献[16-19]通过增加积分项校正了梯度下降时固定步长引起的误差,使其快速收敛到全局成本函数的最优值点,其本质都是利用分布式比例-积分(proportional-integral, PI)控制策略.具体地,文献[16]提出了分布式PI优化算法,该算法的设计运用了反馈控制的思想;文献[17]在文献[16]的基础上,提出了分布式PI优化算法的离散化形式,并推广到时变连通图上;文献[18]用分布式PI优化算法解决了带有等式和不等式约束的分布式优化问题;文献[19]提出了两种基于事件触发的分布式PI优化算法,减少了通讯负担.

上述文献提出的算法都是渐近或指数收敛的,因此收敛时间无法精确确定.然而,在很多实际问题中,例如移动机器人的协同调度^[20]、分布式智能电网的经济调度^[21]等问题,往往需要算法在一定时间内快速收敛.针对该问题,目前研究的分布式优化算法主要是基于有限/固定时间理论.例如,文献[22]提出了一种基于有限时间收敛的分布式算法求解带等式约束的分布式优化问题.文献[23]提出了一种有限时间收敛的分布式PI优化算法.文献[24]基于固定时间理论,从滑膜控制的角度提出了一种分布式优化算法.然而,基于有限时间理论的算法收敛时间与系统的初值有关^[25],基于固定时间理论的算法收敛时间虽然与系统初值无关,但它依赖于系统的某些参数,所以该方法只能给出收敛时间的上界,不能轻易设计系统的收敛时间^[26].最近,文献[26]提出了一种基于预设时间收敛的分布式优化算法,该算法通过引入时基发生器(time-based generator, TBG),使得算法的收敛时间与系统的初值和参数无关,且可任意预先设计.虽然基于预设时间理论的算法在预设时间内只收敛到全局成本函数的最优值点的邻域内,但因其收敛时间可任意预设,已经引起了学者们的注意,逐渐被拓展到非线性系统^[27]、异构多智能体^[28]和多目标优化^[29-30]等领域.

上述基于预设时间理论^[26-30]的分布式优化算法都是基于局部成本函数强凸假设条件的.然而实际上,该假设往往难以满足.比如,在高维统计分析中,存在一些只关于最优值点强凸的损失函数^[31].为描述这些损失函数,文献[31]提出了有限强凸的概念.近年来,学者们在全局成本函数有限强凸条件下,分别证明了一阶优化算法^[19]、二阶优化算法^[32]和稀

疏优化算法^[33]的指数收敛性.这需要考虑在全局成本函数有限强凸的条件下,预设时间收敛的分布式PI优化算法的设计.

本文的主要工作如下:首先,通过引入TBG技术,提出两种预设时间收敛的分布式PI优化算法.其次,利用Lyapunov理论证明在全局成本函数为有限强凸、局部成本函数为可微凸函数,且具有局部Lipschitz梯度的条件下,所提出两种算法实现了预设时间收敛,即在任意预设时间内先收敛到全局成本函数全局最优值点的邻域内,然后精确收敛到全局最优值点.与文献[16-17,26,29]在局部成本函数强凸假设下的研究相比,本文只要求全局成本函数有限强凸,假设更加放松,适用性更强.与文献[21-25,34-35]采用的有限时间和固定时间收敛算法相比,本文算法的收敛时间可以提前任意设计,不依赖于系统的初值和参数.与文献[36]提出的预设时间收敛的DGD算法相比,本文算法利用历史信息校正了固定步长的误差,收敛精度更高.

1 基础概念

本节介绍一些符号的定义,以及有关图论、矩阵论和预设时间收敛的相关知识.

1.1 符号说明

定义 \mathbf{R} 为实数集合, \mathbf{R}^+ 为正实数集合, \mathbf{R}^n 为 n 维实向量空间.给定矩阵 A ,则 A^T 表示其转置矩阵,定义符号 $\|\cdot\|$ 为矩阵的诱导2范数或向量的欧几里德范数, $\lambda_2(\cdot)$ 表示对称矩阵的最小正特征值, $A \otimes B$ 表示矩阵 A 和 B 的Kronecker积, $\nabla f(x)$ 表示可微函数 $f(x)$ 的梯度, $\mathbf{1}_n(\mathbf{0}_n)$ 表示全为1(0)的 n 维列向量, I_n 表示 $n \times n$ 的单位阵.给定两个矩阵 M, N ,若 $M \geq N$,则表示 $M - N$ 是半正定的. $\text{diag}(s_1, \dots, s_n)$ 表示由一串实数 s_1, \dots, s_n 构成的对角矩阵.定义向量组 $x_1, \dots, x_N \in \mathbf{R}^n$,则 $\text{col}(x_1, \dots, x_N) = [x_1^T, \dots, x_N^T]^T$.

1.2 图论

本节将介绍图论的一些相关知识^[37].考虑一个包含 N 个智能体(节点)的无向图, $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \mathcal{A})$,其中: $\mathcal{V} = \{1, 2, \dots, N\}$ 为图的节点集合, $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$ 为图 \mathcal{G} 的通讯链路集合, \mathcal{A} 为加权邻接矩阵, $\mathcal{A} = \mathcal{A}^T = [a_{ij}]$.当且仅当节点 i 和节点 j 可以相互通讯时, $a_{ij} > 0$ 且 $(i, j) \in \mathcal{E}$,其中 $i \neq j$.令 $\mathcal{N}_i = \{j \in \mathcal{V} \mid (i, j) \in \mathcal{E}, a_{ij} > 0\}$ 为节点 i 的邻居节点集合, $\text{deg}_i = \sum_{j \in \mathcal{N}_i} a_{ij}$ 为节点 i 的度,则图 \mathcal{G} 的度矩阵定义为 $D = \text{diag}(\text{deg}_i)$,

..., deg_n). Laplacian矩阵定义为 $L = D - A$.

1.3 预设时间收敛

为实现预设时间收敛,现考虑一个连续可微函数 $\xi(t)$ ^[38] 满足如下条件:

$$\begin{cases} \xi(t) = 0, t = 0; \\ \xi(t) = 1, t \geq t_f; \end{cases} \quad (1a)$$

$$\begin{cases} \dot{\xi}(t) = 0, t = 0 \text{ or } t \geq t_f; \\ \dot{\xi}(t) > 0, \forall t \in (0, t_f). \end{cases} \quad (1b)$$

其中: t_f 表示预设时间,是一个可以任意小的正数. 函数 $\xi(t)$ 被称为TBG,以下系统应用TBG实现预设时间收敛.

引理1 考虑如下系统^[26]:

$$\dot{\chi}(t) = -\vartheta k(t)\chi(t), \chi(0) = \chi_0. \quad (2)$$

其中: 常数 $\vartheta > 0$, $\chi(t)$ 为该系统在 t 时刻的状态. 令时变增益 $k(t)$ 满足如下条件:

$$k(t) = \frac{\dot{\xi}(t)}{1 - \xi(t) + \sigma} \geq 0. \quad (3)$$

其中: σ 是满足 $0 < \sigma \ll 1$ 的常数, $\xi(t)$ 如式(1)所示. 对式(2)关于时间 $(0, t_f)$ 进行积分^[26], 可得

$$\chi(t) = \left(1 - \frac{\xi(t)}{1 + \sigma}\right)^\vartheta \chi_0. \quad (4)$$

根据式(1)可得,选择合适的参数 σ 和 ϑ , 对于 $\forall t_f \in \mathbf{R}^+$, 都可保证 $\chi(t_f)$ 收敛到一个常数 $(\sigma/(1 + \sigma))^\vartheta \chi_0 \ll 1$. 由此可知,系统的收敛时间 t_f 可以提前任意设计,不依赖于系统的初值和参数.

下面根据文献[29]给出连续系统在 t_f 时间内预设时间收敛的定义.

定义1 针对式(2)所示的系统,对于任意的初始状态 $x(0)$,如果系统满足下列条件:

$$\lim_{t \rightarrow t_f} \|\chi(t)\| \leq c, \quad (5a)$$

$$\|\chi(t)\| \leq c, \forall t > t_f, \quad (5b)$$

$$\lim_{t \rightarrow \infty} \|\chi(t)\| = 0, \quad (5c)$$

则称该系统实现预设时间收敛. 其中 c 是满足 $0 < c \ll 1$ 的常数.

2 问题描述

考虑由 N 个智能体组成的多智能体系统,各智能体之间通过潜在的无向图 $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \mathcal{A})$ 构成的网络进行通讯. 其中每个智能体都有一个局部凸函数 $f_i(x) : \mathbf{R}^n \rightarrow \mathbf{R}, i \in \mathcal{V}$. 本文所研究问题的目标是,通过智能体和邻居节点的局部信息协同交互,使所有局部成本函数 $f_i(x)$ 之和构成的全局成本函数 $\tilde{f}(x)$

最小,即

$$\min_{x \in \mathbf{R}^n} \tilde{f}(x) = \sum_{i=1}^N f_i(x). \quad (6)$$

令 $x^* = \arg \min_{x \in \mathbf{R}^n} \tilde{f}(x)$, 即 x^* 为全局成本函数 $\tilde{f}(x)$ 的全局最优值点.

下面给出一些必要的假设.

假设1 无向图 \mathcal{G} 是连通的.

假设2 对于每个 $i \in \mathcal{V}$, 局部成本函数 $f_i(x)$ 在紧集 $D \subseteq \mathbf{R}^n$ 中,具有局部Lipschitz梯度,即存在常数 $\mathcal{L}_{f_i}(D) > 0$, 对 $\forall x, y \in D$ 满足

$$\|\nabla f_i(x) - \nabla f_i(y)\| \leq \mathcal{L}_{f_i}(D) \|x - y\|, \quad (7)$$

其中 $\mathcal{L}_{f_i}(D)$ 称为 $f_i(x)$ 在紧集 D 的Lipschitz常数.

假设3 局部成本函数 $f_i(x) (i \in \mathcal{V})$ 是连续可微凸函数,且全局成本函数 $\tilde{f}(x)$ 的最优值是有界的.

假设4 全局成本函数 $\tilde{f}(x)$ 关于全局最优值点 x^* 是 m_f^- 有限强凸的,即存在常数 $m_f > 0$, 对于任意的 $x \in \mathbf{R}^n$ 下式成立^[13,32]:

$$\sum_{i=1}^N (\nabla f_i(x) - \nabla f_i(x^*))^T (x - x^*) \geq m_f \|x - x^*\|^2.$$

注1 假设1保证每个智能体的局部信息都能通过信息交互分散至整个网络. 假设2是对局部成本函数光滑性的假设,被广泛应用于分布式优化算法中. 在假设3的条件下, $\tilde{f}(x)$ 的全局最优值点 x^* 存在,但可能不唯一. 假设4中的“有限”是指 $\tilde{f}(x)$ 只关于全局最优值点 x^* 强凸,而不需要在其他点强凸. 假设4中有限强凸的条件保证了全局最优值点 x^* 的唯一性. 同时,区别于文献[16-17,26,29]要求局部成本函数强凸,本文只要求全局成本函数有限强凸,放松了假设条件,因此适用性更广.

根据上述假设和文献[32,39],得到以下引理.

引理2 在无向连通图 \mathcal{G} 中, Laplacian矩阵是半正定的,如果令

$$K_N = I_N - \frac{1}{N} \mathbf{1}_N \mathbf{1}_N^T, \quad (8)$$

则

$$K_N L = L K_N = L, K_N^T K_N = K_N,$$

$$0 \leq \lambda_2(L) K_N \leq L.$$

引理3 在无向连通图 \mathcal{G} 中,存在正交矩阵 $[r \ R] \in \mathbf{R}^{N \times N}$, 其中 $r = \frac{1}{\sqrt{N}} \mathbf{1}_N$ 且 $R \in \mathbf{R}^{N \times (N-1)}$, 令 $P = R(A_1)^{-1} R^T$, 若

$$L = [r \ R] \begin{bmatrix} 0 & \\ & A_1 \end{bmatrix} \begin{bmatrix} r^T \\ R^T \end{bmatrix}, \quad (9)$$

则可得 $PL = LP = K_N, P \leq \frac{K_N}{\lambda_2(L)}$. 其中, $\Lambda_1 = \text{diag}(\lambda_2(L), \dots, \lambda_n(L)), 0 < \lambda_2(L) \leq \dots \leq \lambda_n(L)$ 为 Laplacian 矩阵 L 的特征值.

引理4 若假设1~假设4成立,对于问题(6),在紧集 $D \subseteq \mathbf{R}^n$ 中,对于任意 $r \geq 0$,则有

$$(\nabla \tilde{f}(\mathbf{x}) - \nabla \tilde{f}(\mathbf{x}^*))^T (\mathbf{x} - \mathbf{x}^*) + r \mathbf{x}^T (L \otimes I_n) \mathbf{x} \geq m(r) \|\mathbf{x} - \mathbf{x}^*\|^2, \forall \mathbf{x} \in \mathbf{R}^{Nn}. \quad (10)$$

其中

$$\begin{aligned} \tilde{f}(\mathbf{x}) &= \sum_{i=1}^N f_i(x_i), \mathbf{x} = \text{col}(x_1, \dots, x_N), \\ \mathbf{x}^* &= \mathbf{1}_N \otimes x^*, \\ m(r) &= \min \left\{ m_f - 2\mathcal{L}_f(D)l, \frac{r\lambda_2(L)l^2}{1+l^2} \right\}, \\ \mathcal{L}_f(D) &= \max_{i \in \mathcal{V}} \mathcal{L}_{f_i}(D), l \in \left(0, \frac{m_f}{2\mathcal{L}_f(D)} \right). \end{aligned}$$

3 基于预设时间收敛的分布式PI优化算法

本文在全局成本函数 $\tilde{f}(x)$ 有限强凸的条件下,研究分布式优化问题. 针对问题(6),为了保证算法在预设时间内收敛,利用TBG技术,引入时变增益 $k(t)$,提出两种基于预设时间收敛的分布式PI优化算法,即预设时间收敛的ZISP-PI算法(zero initial integral sum predefined-time distributed proportional-integral algorithm)和AIP-PI算法(arbitrary initial integral predefined-time distributed proportional-integral algorithm),并给出收敛性分析.

3.1 ZISP-PI 算法

本节将提出ZISP-PI算法及定理,并对算法的收敛性进行分析.

ZISP-PI算法如下:

$$\begin{aligned} \dot{x}_i(t) &= (k(t) + 1) \left[- \sum_{j=1}^N L_{ij} x_j(t) - q_i(t) - \nabla f_i(x_i(t)) \right], \forall x_i(0) \in \mathbf{R}^n; \quad (11a) \end{aligned}$$

$$\dot{q}_i(t) = (k(t) + 1) \sum_{j=1}^N L_{ij} x_j(t), \sum_{i=1}^N q_i(0) = \mathbf{0}_n. \quad (11b)$$

该算法采用比例积分反馈机制. 式(11a)和(11b)分别是多智能体系统对状态估计和积分信息的更新策略. 具体而言,通过时变增益 $k(t) + 1$ 保证系统在预设时间内收敛,其中 $k(t) = \frac{\dot{\xi}(t)}{1 - \xi(t) + \sigma} \geq 0, \xi(t)$ 满足式(1),式(11a)中的 $\sum_{j=1}^N L_{ij} x_j(t)$ 是为了确保智能体

之间的通讯,保持智能体的一致性; $-\nabla f_i(x_i(t))$ 为局部成本函数的梯度下降项,保证每个智能体的估计以当前的梯度下降方向进行更新. 由于采用了固定步长,系统往往不能精确收敛到全局最优值点,因此引入积分项 $q_i(t)$,用于校正固定步长所产生的误差.

接下来,对如式(11)所示的ZISP-PI算法进行收敛性分析,得到如下定理.

定理1 若假设1~假设4成立,对于每一个智能体运用本文提出的ZISP-PI算法求解分布式优化问题(6),则可以实现预设时间收敛,即

$$\lim_{t \rightarrow t_f} \|x_i(t) - x^*\| \leq \sqrt{\frac{2}{r_1} \left(\frac{\sigma}{1 + \sigma} \right)^{\frac{r_4}{r_2}} V_1(0)}; \quad (12a)$$

$$\|x_i(t) - x^*\| \leq \sqrt{\frac{2}{r_1} \left(\frac{\sigma}{1 + \sigma} \right)^{\frac{r_4}{r_2}} V_1(0)}, \forall t > t_f; \quad (12b)$$

$$\lim_{t \rightarrow \infty} \|x_i(t) - x^*\| = 0. \quad (12c)$$

各参数的取值如下:

$$r_1 > 0, \quad (13a)$$

$$r_2 = \max \left\{ \frac{r_1 + 1}{2}, \frac{r_1 + 1 + 3\lambda_2(L)}{2\lambda_2(L)} \right\} > 0, \quad (13b)$$

$$r_3 = \frac{r_1}{r_1 + 1} > 0, \quad (13c)$$

$$r_4 = \min \left\{ m_1(r_1 + 1) + \mathcal{L}_f(D)^2, \frac{3}{4} \right\} > 0, \quad (13d)$$

$$m_1 = \min \left\{ m_f - 2\mathcal{L}_f(D)l, \frac{r_3\lambda_2(L)l^2}{1+l^2} \right\} > 0. \quad (13e)$$

其中: V_1 为Lyapunov 候选函数, $V_1(0)$ 为其初始状态值, $\mathcal{L}_f(D) = \max_{i \in \mathcal{V}} \mathcal{L}_{f_i}(D), l \in \left(0, \frac{m_f}{2\mathcal{L}_f(D)} \right)$.

证明见附录A.

注2 如式(12)所示的定理1满足如式(5)所示的定义1. 具体地,式(12a)表示当 $t \rightarrow t_f$ 时, $\|x_i(t) - x^*\|$ 收敛到 $\left(0, \sqrt{\frac{2}{r_1} \left(\frac{\sigma}{1 + \sigma} \right)^{\frac{r_4}{r_2}} V_1(0)} \right)$ 内;式(12b)表示当 $t > t_f$ 时, $\|x_i(t) - x^*\|$ 也在该区间内;式(12c)表示当 $t \rightarrow \infty$ 时, $x_i(t)$ 精确收敛到唯一的全局最优值点 x^* . 值得注意的是,收敛上界 $\sqrt{\frac{2}{r_1} \left(\frac{\sigma}{1 + \sigma} \right)^{\frac{r_4}{r_2}} V_1(0)}$ 与 t_f 无关,即无论 t_f 取多小, $x_i(t)$ 的收敛上界不变.

3.2 AIP-PI算法

第3.1节所提出的算法需要满足 $\sum_{i=1}^N q_i(0) = \mathbf{0}_n$ 的初始条件,而在一些实际应用中,该条件往往难以满足. 在文献[16,19]的启发下,本文又提出了AIP-PI算法,这种分布式优化算法对积分项 $q_i(0)$ 没有限制.

AIP-PI算法与ZISP-PI算法相类似,都采用比例积分策略.但是,AIP-PI算法在 $\dot{x}_i(t)$ 更新过程中增加了对积分项 $q_i(t)$ 的通讯,即

$$\dot{x}_i(t) = (k(t) + 1) \left[- \sum_{j=1}^N L_{ij} x_j(t) - \sum_{j=1}^N L_{ij} q_j(t) - \nabla f_i(x_i(t)) \right], \forall x_i(0) \in \mathbf{R}^n, \quad (14a)$$

$$\dot{q}_i(t) = (k(t) + 1) \sum_{j=1}^N L_{ij} x_j(t), \forall q_i(0) \in \mathbf{R}^n. \quad (14b)$$

注3 ZISP-PI算法和AIP-PI算法都采用了PI控制策略和TBG技术,均能实现预设时间收敛,但两种算法的优势略有不同. AIP-PI 算法的初值 $q_i(0)$ 均可以任意选取,而ZISP-PI算法要求 $\sum_{i=1}^N q_i(0) = \mathbf{0}_n$,所以AIP-PI算法的要求更少.但ZISP-PI算法不需要与邻居智能体交互积分项 q_i 的信息,因此需要的通讯开销更小.

与定理1类似,对如式(14)所示的AIP-PI算法进行收敛性分析,得到如下定理.

定理2 若假设1~假设4成立,对于每一个智能体采用本文提出的AIP-PI算法求解分布式优化问题(6),则可以实现预设时间收敛,即

$$\lim_{t \rightarrow t_f} \|x_i(t) - x^*\| \leq \sqrt{\frac{2}{\gamma_1} \left(\frac{\sigma}{1 + \sigma} \right)^{\frac{22}{73}} V_2(0)}; \quad (15a)$$

$$\|x_i(t) - x^*\| \leq \sqrt{\frac{2}{\gamma_1} \left(\frac{\sigma}{1 + \sigma} \right)^{\frac{22}{73}} V_2(0)}, \forall t > t_f; \quad (15b)$$

$$\lim_{t \rightarrow \infty} \|x_i(t) - x^*\| = 0. \quad (15c)$$

各参数的取值如下:

$$\gamma_1 = \frac{2\mathcal{L}_f(D)^2}{m_2 \lambda_2(L)} > 0, \quad (16a)$$

$$\gamma_2 = \min \left\{ \frac{\lambda_2(L)}{4}, \gamma_1 m_2, 1 \right\} > 0, \quad (16b)$$

$$\gamma_3 = \frac{\gamma_1 + 3}{2} > 0, \quad (16c)$$

$$m_2 = \min \left\{ m_f - 2\mathcal{L}_f(D)l, \frac{\lambda_2(L)l^2}{2(1+l^2)} \right\}. \quad (16d)$$

其中: V_2 为Lyapunov候选函数, $V_2(0)$ 为其初始状态值.

证明见附录B.

4 仿真实验

本节将考虑 $N = 4$ 的多智能体系统,通讯拓扑图 G 如图1所示.

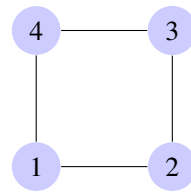


图1 通讯拓扑图

智能体1~智能体4分别对应局部成本函数 $f_1(x) \sim f_4(x)$,实验选取的局部成本函数如下所示:

$$f_1(x) = \frac{1}{2}x^2 - 6, \quad f_2(x) = \frac{1}{2}e^{-\frac{1}{2}x},$$

$$f_3(x) = (x - 2)^2 + e^{-0.1x}, \quad f_4(x) = x^2 \log(2 + x^2).$$

其中:所有局部成本函数均为可微的凸函数且具有Lipschitz梯度, f_1 和 f_3 为强凸函数, f_2 和 f_4 为非强凸函数,全局成本函数 $\sum_{i=1}^4 f_i(x)$ 为有限强凸函数.

考虑预设时间 $t_f = 0.2$ s,在相同的通讯拓扑图和局部成本函数的情况下,将本文提出的ZISP-PI和AIP-PI算法分别与文献[19]中的算法(2)和算法(3)进行比较,初始状态取值如表1所示. $\tau(t)$ 的动态变化如图2所示,结果表明本文所提出算法在预设时间0.2s时收敛到 10^{-20} 数量级,随后收敛到 10^{-47} 数量级,加快了文献[19]中对应算法的收敛速率.对比可得,本文所提出的两种算法在预设时间 t_f 内,能迅速收敛到全局最优值点的邻域,随后收敛到全局最优值点 x^* ,即实现预设时间收敛.

表1 初始状态取值

算法	初值 $x_i(0)$	初值 $q_i(0), i \in \mathcal{V}$
ZISP-PI	(1, 7, 3, -1)	(0.2, 0.6, -0.2, -0.6)
AIP-PI	(1, 7, 3, -1)	(0.5, 0.5, -0.2, -0.5)
文献[19]中算法(2)	(1, 7, 3, -1)	(0.2, 0.6, -0.2, -0.6)
文献[19]中算法(3)	(1, 7, 3, -1)	(0.5, 0.5, -0.2, -0.5)

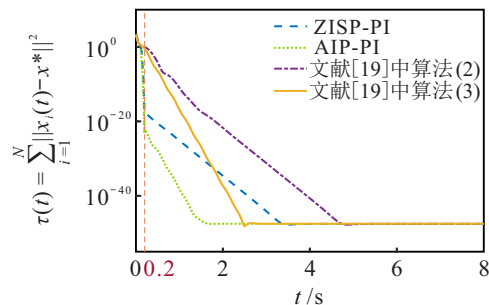


图2 本文算法与文献[19]算法对比

考虑预设时间 $t_f = 0.1$ s,构造TBG如下所示:

$$\xi(t) = \begin{cases} \frac{10}{0.1^6}t^6 - \frac{24}{0.1^5}t^5 + \frac{15}{0.1^4}t^4, & 0 \leq t \leq t_f; \\ 1, & t > t_f. \end{cases} \quad (17)$$

针对ZISP-PI算法,图3展示了智能体状态的演化过程,其中每个智能体在0.1s时收敛到全局最优值点 $x^* = 0.79$ 的邻域内且达到一致.由此可得,改变预设时间 t_f ,仍可以使本文所提出算法在预设时间收敛,验证了本文所提出算法是有效的.

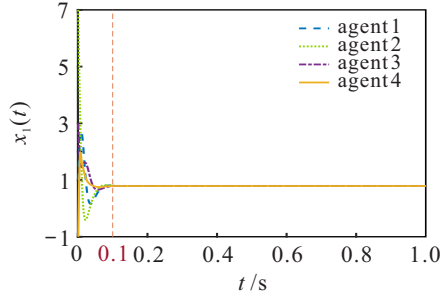


图3 预设时间0.1s时 $x_i(t)$ 的演化曲线

5 结论

针对无向连通图,本文在全局成本函数为有限强凸、局部成本函数为可微凸函数,且Lipschitz连续的条件下,利用TBG技术,提出了两种基于预设时间收敛的分布式优化算法,即ZISP-PI算法和AIP-PI算法.其次,通过Lyapunov理论,证明了本文所提出算法可以实现预设时间收敛.特别地,预设时间的设计不依赖于系统的初始状态和参数,可以任意设计.最后,仿真实验验证了两种算法的有效性.未来的研究方向是将此算法推广到非凸领域.

附录A 定理1的证明.

证明 证明分3步:首先证明全局目标函数在平衡点处达到全局最优;然后构造Lyapunov候选函数;最后证明该算法在预先设计的时间范围内达到定理1所示的预设时间收敛的效果.具体的证明过程如下.

step 1: 为方便起见,将ZISP-PI算法写成紧凑形式,即

$$\dot{x} = (k(t) + 1)[-(L \otimes I_n)x - q - \nabla \tilde{f}(x)], \quad (A1a)$$

$$\dot{q} = (k(t) + 1)(L \otimes I_n)x, \quad q(0) = \mathbf{0}_{Nn}. \quad (A1b)$$

其中: $x = \text{col}(x_1, \dots, x_N) \in \mathbf{R}^{Nn}$, $q = \text{col}(q_1, \dots, q_N) \in \mathbf{R}^{Nn}$, $\nabla \tilde{f}(x) = \text{col}(\nabla f_1(x_1), \dots, \nabla f_N(x_N)) \in \mathbf{R}^{Nn}$.

将式(A1b)两边左乘 $\mathbf{1}_N^T \otimes I_n$,由Laplacian矩阵的性质,可得 $\sum_{i=1}^N \dot{q}_i = \mathbf{0}_n$,因此有

$$\sum_{i=1}^N q_i(t) = \sum_{i=1}^N q_i(0) = \mathbf{0}_n, \quad \forall t \geq 0. \quad (A2)$$

假设式(18)平衡点处的状态估计为 $\bar{x} = \mathbf{1}_N \otimes \theta$,其中 $\theta \in \mathbf{R}^n$,有

$$-(L \otimes I_n)\bar{x} - \bar{q} - \nabla \tilde{f}(\bar{x}) = \mathbf{0}_{Nn}, \quad (A3a)$$

$$(L \otimes I_n)\bar{x} = \mathbf{0}_{Nn}. \quad (A3b)$$

进而可得

$$\bar{q} = -\nabla \tilde{f}(\bar{x}). \quad (A4)$$

将式(A4)两边左乘 $\mathbf{1}_N^T \otimes I_n$,结合式(A2),在平衡点处有 $\sum_{i=1}^N \nabla f_i(x_i) = \sum_{i=1}^N \nabla f_i(\theta) = \mathbf{0}_n$.又因为 $\tilde{f}(x^*) = \sum_{i=1}^N \nabla f_i(x^*) = \mathbf{0}_n$,所以 $\theta = x^*$,即 $\bar{x} = x^*$,其中 $x^* = \mathbf{1}_N \otimes x^*$.

另外,由式(8)和(A2)可得

$$(K_N \otimes I_n)^T(q - \bar{q}) = q - \bar{q}, \quad (A5)$$

由式(A2)和 $\bar{x} = \mathbf{1}_N \otimes \theta$ 可得

$$\bar{x}^T(q - \bar{q}) = 0. \quad (A6)$$

step 2: 考虑如下函数:

$$W_1(x, q) = \frac{r_1 + 1}{2} \|x - \bar{x}\|^2 + \frac{r_1 + 1}{2} (q - \bar{q})^T (P \otimes I_n) (q - \bar{q}), \quad (A7)$$

$$W_2(x, q) = (x - \bar{x})^T (K_N \otimes I_n) (q - \bar{q}) + (q - \bar{q})^T (K_N \otimes I_n) (q - \bar{q}), \quad (A8)$$

其中 r_1 取值如式(13a)所示.

将 $W_1(x, q)$ 沿算法(A1)的方向求导数,可得

$$\begin{aligned} \dot{W}_1(x, q) = & (k(t) + 1)(r_1 + 1)\{(x - \bar{x})^T [-(L \otimes I_n)x - q - \nabla \tilde{f}(x)] + (r_1 + 1)(q - \bar{q})^T (K_N \otimes I_n)x\} = \\ & (k(t) + 1)(r_1 + 1)\{-x^T (L \otimes I_n)x - (x - \bar{x})^T (q - \bar{q}) - (x - \bar{x})^T (\nabla \tilde{f}(x) - \nabla \tilde{f}(\bar{x})) + (q - \bar{q})^T (K_N \otimes I_n)x\} = \\ & (k(t) + 1)(r_1 + 1)\{-x^T (L \otimes I_n)x - (x - \bar{x})^T (q - \bar{q}) + (q - \bar{q})^T x - (x - \bar{x})^T (\nabla \tilde{f}(x) - \nabla \tilde{f}(\bar{x}))\} = \\ & (k(t) + 1)(r_1 + 1)\{-x^T (L \otimes I_n)x - (x - \bar{x})^T (\nabla \tilde{f}(x) - \nabla \tilde{f}(\bar{x}))\}. \end{aligned} \quad (A9)$$

其中:第1个等式是将算法(A1)代入,且应用引理3中的 $PL = K_N$ 所得,第2个等式是应用式(A3b)和(A4)所得,第3个等式是应用式(A5)所得,第4个等式是应用式(A6)所得.

将 $W_2(\mathbf{x}, \mathbf{q})$ 沿算法(A1)的方向求导数, 可得

$$\begin{aligned} \dot{W}_2(\mathbf{x}, \mathbf{q}) = & (k(t) + 1)\{(\mathbf{q} - \bar{\mathbf{q}})^T(K_N \otimes I_n)[-(L \otimes I_n)\mathbf{x} - \\ & \mathbf{q} - \nabla \tilde{f}(\mathbf{x})] + (\mathbf{x} - \bar{\mathbf{x}})^T(K_N \otimes I_n)(L \otimes I_n)\mathbf{x} + \\ & (\mathbf{q} - \bar{\mathbf{q}})^T(K_N \otimes I_n)(L \otimes I_n)\mathbf{x}\} = \\ & (k(t) + 1)\{-(\mathbf{q} - \bar{\mathbf{q}})^T(L \otimes I_n)\mathbf{x} - \\ & (\mathbf{q} - \bar{\mathbf{q}})^T(K_N \otimes I_n)(\mathbf{q} - \bar{\mathbf{q}}) - \\ & (\mathbf{q} - \bar{\mathbf{q}})^T(K_N \otimes I_n)(\nabla \tilde{f}(\mathbf{x}) - \nabla \tilde{f}(\bar{\mathbf{x}})) + \\ & (\mathbf{x} - \bar{\mathbf{x}})^T(L \otimes I_n)\mathbf{x} + (\mathbf{q} - \bar{\mathbf{q}})^T(L \otimes I_n)\mathbf{x}\} \leq \\ & (k(t) + 1)\left\{- (\mathbf{q} - \bar{\mathbf{q}})^T(K_N \otimes I_n)(\mathbf{q} - \bar{\mathbf{q}}) + \right. \\ & \left. \frac{1}{4}(\mathbf{q} - \bar{\mathbf{q}})^T(K_N \otimes I_n)(\mathbf{q} - \bar{\mathbf{q}}) + \right. \\ & \left. \|\nabla \tilde{f}(\mathbf{x}) - \nabla \tilde{f}(\bar{\mathbf{x}})\|^2 + \mathbf{x}^T(L \otimes I_n)\mathbf{x}\right\} \leq \\ & (k(t) + 1)\left\{- \frac{3}{4}(\mathbf{q} - \bar{\mathbf{q}})^T(\mathbf{q} - \bar{\mathbf{q}}) + \right. \\ & \left. \mathcal{L}_f(D)^2\|\mathbf{x} - \bar{\mathbf{x}}\|^2 + \mathbf{x}^T(L \otimes I_n)\mathbf{x}\right\}. \quad (\text{A10}) \end{aligned}$$

其中: 第2个等式是应用引理2的性质 $K_N L = L$ 和式(A4)所得, 第1个不等式是应用Cauchy-Schwarz不等式和式(A3b)所得, 第2个不等式是应用式(7)和(A5)所得.

考虑如下Lyapunov候选函数:

$$V_1(\mathbf{x}, \mathbf{q}) = W_1(\mathbf{x}, \mathbf{q}) + W_2(\mathbf{x}, \mathbf{q}). \quad (\text{A11})$$

下面对 $V_1(\mathbf{x}, \mathbf{q})$ 进行分析.

$$\begin{aligned} V_1(\mathbf{x}, \mathbf{q}) = & W_1(\mathbf{x}, \mathbf{q}) + W_2(\mathbf{x}, \mathbf{q}) = \\ & \frac{r_1}{2}\|\mathbf{x} - \bar{\mathbf{x}}\|^2 + \frac{r_1 + 1}{2}(\mathbf{q} - \bar{\mathbf{q}})^T(P \otimes I_n)(\mathbf{q} - \bar{\mathbf{q}}) + \\ & \frac{1}{2}\|\mathbf{x} - \bar{\mathbf{x}}\|^2 + \frac{1}{2}(\mathbf{q} - \bar{\mathbf{q}})^T(K_N \otimes I_n)(\mathbf{q} - \bar{\mathbf{q}}) + \\ & (\mathbf{x} - \bar{\mathbf{x}})^T(K_N \otimes I_n)(\mathbf{q} - \bar{\mathbf{q}}) + \\ & \frac{1}{2}(\mathbf{q} - \bar{\mathbf{q}})^T(K_N \otimes I_n)(\mathbf{q} - \bar{\mathbf{q}}) \geq \\ & \frac{r_1}{2}\|\mathbf{x} - \bar{\mathbf{x}}\|^2 + \frac{r_1 + 1}{2}(\mathbf{q} - \bar{\mathbf{q}})^T(P \otimes I_n)(\mathbf{q} - \bar{\mathbf{q}}) + \\ & \frac{1}{2}\|\mathbf{q} - \bar{\mathbf{q}}\|^2 \geq \frac{r_1}{2}\|\mathbf{x} - \bar{\mathbf{x}}\|^2. \quad (\text{A12}) \end{aligned}$$

其中: 第1个不等式是应用Cauchy-Schwarz不等式以及式(A5)所得, 第2个不等式成立是因为 $P \otimes I_n$ 是半正定的.

$$\begin{aligned} V_1(\mathbf{x}, \mathbf{q}) = & W_1(\mathbf{x}, \mathbf{q}) + W_2(\mathbf{x}, \mathbf{q}) = \\ & \frac{r_1 + 1}{2}\|\mathbf{x} - \bar{\mathbf{x}}\|^2 + \frac{r_1 + 1}{2}(\mathbf{q} - \bar{\mathbf{q}})^T(P \otimes I_n)(\mathbf{q} - \bar{\mathbf{q}}) + \\ & (\mathbf{x} - \bar{\mathbf{x}})^T(K_N \otimes I_n)(\mathbf{q} - \bar{\mathbf{q}}) + \end{aligned}$$

$$\begin{aligned} & (\mathbf{q} - \bar{\mathbf{q}})^T(K_N \otimes I_n)(\mathbf{q} - \bar{\mathbf{q}}) \leq \\ & \frac{r_1 + 1}{2}\|\mathbf{x} - \bar{\mathbf{x}}\|^2 + \frac{1}{2}\|\mathbf{x} - \bar{\mathbf{x}}\|^2 + \\ & \frac{r_1 + 1}{2\lambda_2(L)}(\mathbf{q} - \bar{\mathbf{q}})^T(K_N \otimes I_n)(\mathbf{q} - \bar{\mathbf{q}}) + \\ & \frac{3}{2}(\mathbf{q} - \bar{\mathbf{q}})^T(K_N \otimes I_n)(\mathbf{q} - \bar{\mathbf{q}}) \leq \\ & \frac{r_1 + 1}{2}\|\mathbf{x} - \bar{\mathbf{x}}\|^2 + \frac{r_1 + 1 + 3\lambda_2(L)}{2\lambda_2(L)}\|\mathbf{q} - \bar{\mathbf{q}}\|^2 \leq \\ & r_2\{\|\mathbf{x} - \bar{\mathbf{x}}\|^2 + \|\mathbf{q} - \bar{\mathbf{q}}\|^2\}. \quad (\text{A13}) \end{aligned}$$

其中: r_2 的取值如式(13b)所示, 第1个不等式是应用Cauchy-Schwarz不等式和引理2中的 $P \leq \frac{1}{\lambda_2(L)}K_N$ 所得, 第2个不等式是应用式(A5)所得.

另外, 由式(A9)和(A10)可得

$$\begin{aligned} \dot{V}_1(\mathbf{x}, \mathbf{q}) = & \dot{W}_1(\mathbf{x}, \mathbf{q}) + \dot{W}_2(\mathbf{x}, \mathbf{q}) = \\ & (k(t) + 1)\left\{- r_1\mathbf{x}^T(L \otimes I_n)\mathbf{x} - \right. \\ & \left. \frac{3}{4}\|\mathbf{q} - \bar{\mathbf{q}}\|^2 + \mathcal{L}_f(D)^2\|\mathbf{x} - \bar{\mathbf{x}}\|^2 - \right. \\ & \left. (r_1 + 1)(\mathbf{x} - \bar{\mathbf{x}})^T(\nabla \tilde{f}(\mathbf{x}) - \nabla \tilde{f}(\bar{\mathbf{x}}))\right\}. \quad (\text{A14}) \end{aligned}$$

由于假设1~假设4成立, 可得

$$\begin{aligned} & (\mathbf{x} - \bar{\mathbf{x}})^T(\nabla \tilde{f}(\mathbf{x}) - \nabla \tilde{f}(\bar{\mathbf{x}})) + r_3\mathbf{x}^T(L \otimes I_n)\mathbf{x} \geq \\ & m_1\|\mathbf{x} - \bar{\mathbf{x}}\|^2, \quad (\text{A15}) \end{aligned}$$

其中 r_3 和 m_1 分别如(13c)和(13e)所示.

由式(A14)和(A15)可得

$$\begin{aligned} \dot{V}_1(\mathbf{x}, \mathbf{q}) \leq & - (k(t) + 1)\left\{(r_1 + 1)m_1\|\mathbf{x} - \bar{\mathbf{x}}\|^2 + \right. \\ & \left. \frac{3}{4}\|\mathbf{q} - \bar{\mathbf{q}}\|^2 + \mathcal{L}_f(D)^2\|\mathbf{x} - \bar{\mathbf{x}}\|^2\right\} \leq \\ & - r_4(k(t) + 1)\{\|\mathbf{x} - \bar{\mathbf{x}}\|^2 + \|\mathbf{q} - \bar{\mathbf{q}}\|^2\}, \quad (\text{A16}) \end{aligned}$$

其中 r_4 如式(13d)所示.

step 3: 由式(A13)和(A16)可得

$$\dot{V}_1(\mathbf{x}(t), \mathbf{q}(t)) \leq -\frac{r_4(k(t) + 1)}{r_2}V_1(\mathbf{x}(0), \mathbf{q}(0)). \quad (\text{A17})$$

接下来对收敛速度进行说明. 由式(A13)和(A17)可得

$$V_1(\mathbf{x}(t), \mathbf{q}(t)) \leq V_1(\mathbf{x}(0), \mathbf{q}(0))e^{-\frac{r_4(k(t)+1)}{r_2}t}.$$

结合式(A12)可得, 当 $0 < t < t_f$ 时, $k(t) > 0$, $x_i(t)$ 以不小于 $\frac{r_4(k(t) + 1)}{2r_2} > 0$ 的速度指数收敛到全局最优值点 x^* 的邻域内; 当 $t \geq t_f$ 时, $k(t) = 0$, $x_i(t)$ 以不小于 $\frac{r_4}{2r_2} > 0$ 的速度指数收敛到全局唯一的最优值点

x^* .

当 $t \rightarrow t_f$ 时, $k(t) = 0$, 由引理 1 和式 (A17) 可得

$$\lim_{t \rightarrow t_f} V_1(\mathbf{x}(t), \mathbf{q}(t)) \rightarrow \left(\frac{\sigma}{1+\sigma}\right)^{\frac{r_4}{r_2}} V_1(\mathbf{x}(0), \mathbf{q}(0)); \tag{A18}$$

由式 (A12) 和 (A18) 可得

$$\lim_{t \rightarrow t_f} \|\mathbf{x} - \bar{\mathbf{x}}\| \leq \sqrt{\frac{2}{r_1} \left(\frac{\sigma}{1+\sigma}\right)^{\frac{r_4}{r_2}} V_1(\mathbf{x}(0), \mathbf{q}(0))}. \tag{A19}$$

当 $t \geq t_f$ 时, 由式 (A12) 和 (A18) 可得

$$\|\mathbf{x} - \bar{\mathbf{x}}\| \leq \sqrt{\frac{2}{r_1} \left(\frac{\sigma}{1+\sigma}\right)^{\frac{r_4}{r_2}} V_1(\mathbf{x}(0), \mathbf{q}(0))}. \tag{A20}$$

由 step 1 可知 $\bar{\mathbf{x}} = \mathbf{x}^*$. 综上, 定理 1 得证. \square

附录 B 定理 2 的证明.

证明 与定理 1 证明相类似, 定理 2 证明过程分为 3 步.

step 1: 将 AIP-PI 算法写成如下紧凑形式:

$$\dot{\mathbf{x}} = -(k(t) + 1)[(L \otimes I_n)\mathbf{x} + (L \otimes I_n)\mathbf{q} + \nabla \tilde{f}(\mathbf{x})], \forall \mathbf{x}(0) \in \mathbf{R}^n, \tag{B1a}$$

$$\dot{\mathbf{q}} = (k(t) + 1)(L \otimes I_n)\mathbf{x}, \forall \mathbf{q}(0) \in \mathbf{R}^n. \tag{B1b}$$

在式 (B1) 平衡点 $(\bar{\mathbf{x}}, \bar{\mathbf{q}})$ 处, 有

$$-(L \otimes I_n)\bar{\mathbf{x}} - (L \otimes I_n)\bar{\mathbf{q}} - \nabla \tilde{f}(\bar{\mathbf{x}}) = \mathbf{0}_{Nn}, \tag{B2a}$$

$$(L \otimes I_n)\bar{\mathbf{x}} = \mathbf{0}_{Nn}. \tag{B2b}$$

其中: $\bar{\mathbf{x}} = \mathbf{1}_N \otimes \theta$ ($\theta \in \mathbf{R}^n$), 进而可得

$$(L \otimes I_n)\bar{\mathbf{q}} = \nabla \tilde{f}(\bar{\mathbf{x}}). \tag{B3}$$

与定理 1 同理, 将式 (B2a) 两边左乘 $\mathbf{1}_N^T \otimes I_n$, 由无向连通图的 Laplacian 矩阵性质可得, 在平衡点处 $\sum_{i=1}^N \nabla f_i(x_i) = \sum_{i=1}^N \nabla f_i(\theta) = \mathbf{0}_n$. 又因为 $\tilde{f}(x^*) = \sum_{i=1}^N \nabla f_i(x^*) = \mathbf{0}_n$, 所以 $\theta = x^*$, 即 $\bar{\mathbf{x}} = \mathbf{x}^*$.

step 2: 考虑如下函数:

$$W_3(\mathbf{x}, \mathbf{q}) = (\mathbf{x} - \bar{\mathbf{x}})^T (K_N \otimes I_n) (\mathbf{q} - \bar{\mathbf{q}}), \tag{B4}$$

$$W_4(\mathbf{x}, \mathbf{q}) = \frac{\gamma_1 + 2}{2} (\mathbf{q} - \bar{\mathbf{q}})^T (K_N \otimes I_n) (\mathbf{q} - \bar{\mathbf{q}}) + \frac{\gamma_1 + 1}{2} \|\mathbf{x} - \bar{\mathbf{x}}\|^2, \tag{B5}$$

其中 γ_1 如式 (16a) 所示.

将式 $\dot{W}_3(\mathbf{x}, \mathbf{q})$ 沿轨迹 (B1) 的方向求导数, 有

$$\dot{W}_3(\mathbf{x}, \mathbf{q}) = (k(t) + 1) \{(\mathbf{q} - \bar{\mathbf{q}})^T (K_N \otimes I_n) [-(L \otimes I_n)\mathbf{x} -$$

$$(L \otimes I_n)\mathbf{q} - \nabla \tilde{f}(\mathbf{x})] + (\mathbf{x} - \bar{\mathbf{x}})^T (L \otimes I_n)\mathbf{x}\} = (k(t) + 1) \{(\mathbf{q} - \bar{\mathbf{q}})^T (K_N \otimes I_n) [-(L \otimes I_n)\mathbf{x} - (L \otimes I_n)(\mathbf{q} - \bar{\mathbf{q}}) - (\nabla \tilde{f}(\mathbf{x}) - \nabla \tilde{f}(\bar{\mathbf{x}}))] + \mathbf{x}^T (L \otimes I_n)\mathbf{x}\} \leq$$

$$(k(t) + 1) \left\{ -\frac{3}{4} (\mathbf{q} - \bar{\mathbf{q}})^T (L \otimes I_n) (\mathbf{q} - \bar{\mathbf{q}}) - (\mathbf{q} - \bar{\mathbf{q}})^T (L \otimes I_n)\mathbf{x} + (\mathbf{x} - \bar{\mathbf{x}})^T (L \otimes I_n)\mathbf{x} + \frac{\lambda_2(L)}{4} (\mathbf{q} - \bar{\mathbf{q}})^T (K_N \otimes I_n) (\mathbf{q} - \bar{\mathbf{q}}) + \frac{1}{\lambda_2(L)} \|\nabla \tilde{f}(\mathbf{x}) - \nabla \tilde{f}(\bar{\mathbf{x}})\|^2 \right\} \leq (k(t) + 1) \left\{ -\frac{1}{2} (\mathbf{q} - \bar{\mathbf{q}})^T (L \otimes I_n) (\mathbf{q} - \bar{\mathbf{q}}) - (\mathbf{q} - \bar{\mathbf{q}})^T (L \otimes I_n)\mathbf{x} + (\mathbf{x} - \bar{\mathbf{x}})^T (L \otimes I_n)\mathbf{x} + \frac{1}{\lambda_2(L)} \|\nabla \tilde{f}(\mathbf{x}) - \nabla \tilde{f}(\bar{\mathbf{x}})\|^2 \right\} \leq (k(t) + 1) \left\{ -\frac{1}{2} (\mathbf{q} - \bar{\mathbf{q}})^T (L \otimes I_n) (\mathbf{q} - \bar{\mathbf{q}}) - (\mathbf{q} - \bar{\mathbf{q}})^T (L \otimes I_n)\mathbf{x} + (\mathbf{x} - \bar{\mathbf{x}})^T (L \otimes I_n)\mathbf{x} + \frac{\mathcal{L}_f(D)^2}{\lambda_2(L)} \|\mathbf{x} - \bar{\mathbf{x}}\|^2 \right\}. \tag{B6}$$

其中: 第 1 个等式是将算法 (B1) 代入, 第 2 个等式是应用式 (B2b) 和 (B3) 所得, 第 1 个不等式是应用引理 2 中的性质 $K_N L = L$ 和 Cauchy-Schwarz 不等式所得, 第 2 个不等式是应用引理 2 中的性质 $\lambda_2(L) K_N \leq L$ 所得, 第 3 个不等式是应用假设 2 的条件 (7) 所得.

类似地, 将 W_4 沿轨迹 (B1) 的方向求导数, 有

$$\dot{W}_4(\mathbf{x}, \mathbf{q}) = (k(t) + 1) \{(\gamma_1 + 1)(\mathbf{x} - \bar{\mathbf{x}})^T [-(L \otimes I_n)\mathbf{x} - (L \otimes I_n)\mathbf{q} - \nabla \tilde{f}(\mathbf{x})] + (\gamma_1 + 1)(\mathbf{q} - \bar{\mathbf{q}})^T (L \otimes I_n)\mathbf{x}\} = (k(t) + 1) \{(\mathbf{q} - \bar{\mathbf{q}})^T (L \otimes I_n)\mathbf{x} - \mathbf{x}^T (L \otimes I_n)\mathbf{x} - \gamma_1 (\mathbf{x} - \bar{\mathbf{x}})^T (L \otimes I_n)(\mathbf{x} - \bar{\mathbf{x}}) - (\gamma_1 + 1)(\mathbf{x} - \bar{\mathbf{x}})^T (\nabla \tilde{f}(\mathbf{x}) - \nabla \tilde{f}(\bar{\mathbf{x}}))\}, \tag{B7}$$

进而可得

$$\dot{W}_4(\mathbf{x}, \mathbf{q}) \leq (k(t) + 1) \left\{ (\mathbf{q} - \bar{\mathbf{q}})^T (L \otimes I_n)\mathbf{x} - \mathbf{x}^T (L \otimes I_n)\mathbf{x} - \frac{\gamma_1}{2} \mathbf{x}^T (L \otimes I_n)\mathbf{x} + \frac{1}{4} (\mathbf{q} - \bar{\mathbf{q}})^T (L \otimes I_n) (\mathbf{q} - \bar{\mathbf{q}}) - (\gamma_1 + 1)(\mathbf{x} - \bar{\mathbf{x}})^T (\nabla \tilde{f}(\mathbf{x}) - \nabla \tilde{f}(\bar{\mathbf{x}})) \right\}. \tag{B8}$$

由于假设 1 ~ 假设 4 成立, 可得

$$(\mathbf{x} - \bar{\mathbf{x}})^T (\nabla \tilde{f}(\mathbf{x}) - \nabla \tilde{f}(\bar{\mathbf{x}})) \geq$$

$$m_2 \|\mathbf{x} - \bar{\mathbf{x}}\|^2 - \frac{1}{2} \mathbf{x}^T (L \otimes I_n) \mathbf{x}. \quad (\text{B9})$$

因此,由 $(\mathbf{x} - \bar{\mathbf{x}})^T (\nabla \tilde{f}(\mathbf{x}) - \nabla \tilde{f}(\bar{\mathbf{x}})) \geq 0$, 以及式 (B6)~(B9) 可得

$$\begin{aligned} \dot{W}_3(\mathbf{x}, \mathbf{q}) + \dot{W}_4(\mathbf{x}, \mathbf{q}) \leq & \\ & - (k(t) + 1) \left[\frac{1}{4} (\mathbf{q} - \bar{\mathbf{q}})^T (L \otimes I_n) (\mathbf{q} - \bar{\mathbf{q}}) + \right. \\ & \left. \gamma_1 m_2 \|\mathbf{x} - \bar{\mathbf{x}}\|^2 \right]. \end{aligned} \quad (\text{B10})$$

考虑如下 Lyapunov 候选函数:

$$V_2(\mathbf{x}, \mathbf{q}) = W_3(\mathbf{x}, \mathbf{q}) + W_4(\mathbf{x}, \mathbf{q}). \quad (\text{B11})$$

由式 (B11) 可得, V_2 沿轨迹 (B1) 的方向导数为

$$\begin{aligned} \dot{V}_2(\mathbf{x}, \mathbf{q}) \leq & \\ & - (k(t) + 1) \left\{ \gamma_1 m_2 \|\mathbf{x} - \bar{\mathbf{x}}\|^2 + \right. \\ & \left. \frac{1}{4} (\mathbf{q} - \bar{\mathbf{q}})^T (L \otimes I_n) (\mathbf{q} - \bar{\mathbf{q}}) \right\} \leq \\ & - (k(t) + 1) \left\{ \gamma_1 m_2 \|\mathbf{x} - \bar{\mathbf{x}}\|^2 + \right. \\ & \left. \frac{\lambda_2(L)}{4} (\mathbf{q} - \bar{\mathbf{q}})^T (K_N \otimes I_n) (\mathbf{q} - \bar{\mathbf{q}}) \right\} \leq \\ & - (k(t) + 1) \gamma_2 \{ \|\mathbf{x} - \bar{\mathbf{x}}\|^2 + \\ & (\mathbf{q} - \bar{\mathbf{q}})^T (K_N \otimes I_n) (\mathbf{q} - \bar{\mathbf{q}}) \}, \end{aligned} \quad (\text{B12})$$

其中 γ_2 如式 (16b) 所示.

令 $\gamma_3 = (\gamma_1 + 3)/2$, 由 Cauchy-Schwarz 不等式可得

$$V_2(\mathbf{x}, \mathbf{q}) \leq -\gamma_3 [(\mathbf{q} - \bar{\mathbf{q}})^T (K_N \otimes I_n) (\mathbf{q} - \bar{\mathbf{q}}) - \|\mathbf{x} - \bar{\mathbf{x}}\|^2]. \quad (\text{B13})$$

下面对 $V_2(\mathbf{x}, \mathbf{q})$ 进行分析.

$$\begin{aligned} V_2(\mathbf{x}, \mathbf{q}) = & \\ & (\mathbf{x} - \bar{\mathbf{x}})^T (K_N \otimes I_n) (\mathbf{q} - \bar{\mathbf{q}}) + \frac{1}{2} \|\mathbf{x} - \bar{\mathbf{x}}\|^2 + \\ & \frac{1}{2} (\mathbf{q} - \bar{\mathbf{q}})^T (K_N \otimes I_n) (\mathbf{q} - \bar{\mathbf{q}}) + \\ & \frac{\gamma_1 + 1}{2} (\mathbf{q} - \bar{\mathbf{q}})^T (K_N \otimes I_n) (\mathbf{q} - \bar{\mathbf{q}}) + \frac{\gamma_1}{2} \|\mathbf{x} - \bar{\mathbf{x}}\|^2 \geq \\ & \frac{\gamma_1 + 1}{2} (\mathbf{q} - \bar{\mathbf{q}})^T (K_N \otimes I_n) (\mathbf{q} - \bar{\mathbf{q}}) + \frac{\gamma_1}{2} \|\mathbf{x} - \bar{\mathbf{x}}\|^2 \geq \\ & \frac{\gamma_1}{2} \|\mathbf{x} - \bar{\mathbf{x}}\|^2. \end{aligned} \quad (\text{B14})$$

其中: 第1个不等式是应用 Cauchy-Schwarz 不等式和引理2中的性质 $K_N^T K_N = K_N$ 所得, 第2个不等式是应用式 (A5) 所得.

step 3: 由式 (B12) 和 (B13) 可得

$$\dot{V}_2(\mathbf{x}(t), \mathbf{q}(t)) \leq -\frac{\gamma_2(k(t) + 1)}{\gamma_3} V_2(\mathbf{x}(0), \mathbf{q}(0)). \quad (\text{B15})$$

下面对收敛速度进行证明. 由式 (B13) 和 (B15) 可得

$$V_2(\mathbf{x}(t), \mathbf{q}(t)) \leq V_2(\mathbf{x}(0), \mathbf{q}(0)) e^{-\frac{\gamma_2(k(t)+1)}{\gamma_3} t}.$$

结合式 (B14) 可得, 当 $0 < t < t_f$ 时, $k(t) > 0$, $x_i(t)$ 以不小于 $\frac{\gamma_2(k(t)+1)}{2\gamma_3} > 0$ 的速度指数收敛到全局最优值点 x^* 的邻域内; 当 $t \geq t_f$ 时, $k(t) = 0$, $x_i(t)$ 以不小于 $\frac{\gamma_2}{2\gamma_3} > 0$ 的速度指数收敛到全局唯一的最优值点 x^* .

当 $t \rightarrow t_f$ 时, $k(t) = 0$. 由引理1和式 (B15) 可得

$$\lim_{t \rightarrow t_f} V_2(\mathbf{x}(t), \mathbf{q}(t)) \rightarrow \left(\frac{\sigma}{1 + \sigma} \right)^{\frac{\gamma_2}{\gamma_3}} V_2(\mathbf{x}(0), \mathbf{q}(0)); \quad (\text{B16})$$

由式 (B14) 和 (B16) 可得

$$\lim_{t \rightarrow t_f} \|\mathbf{x} - \bar{\mathbf{x}}\| \leq \sqrt{\frac{2}{\gamma_1} \left(\frac{\sigma}{1 + \sigma} \right)^{\frac{\gamma_2}{\gamma_3}} V_2(\mathbf{x}(0), \mathbf{q}(0))}. \quad (\text{B17})$$

当 $t \geq t_f$ 时, 由式 (B14) 和 (B15) 可得

$$\|\mathbf{x} - \bar{\mathbf{x}}\| \leq \sqrt{\frac{2}{\gamma_1} \left(\frac{\sigma}{1 + \sigma} \right)^{\frac{\gamma_2}{\gamma_3}} V_2(\mathbf{x}(0), \mathbf{q}(0))}. \quad (\text{B18})$$

由 step 1 可知 $\mathbf{x} = \mathbf{x}^*$. 综上, 定理2得证. \square

参考文献 (References)

- [1] Tsitsiklis J N. Problems in decentralized decision making and computation[D]. Cambridge: Massachusetts Institute of Technology, 1984: 147-152.
- [2] Tsitsiklis J N, Bertsekas D, Athans M. Distributed asynchronous deterministic and stochastic gradient optimization algorithms[J]. IEEE Transactions on Automatic Control, 1986, 31(9): 803-812.
- [3] Yang T, Lu J, Wu D, et al. A distributed algorithm for economic dispatch over time-varying directed networks with delays[J]. IEEE Transactions on Industrial Electronics, 2017, 64(6): 5095-5106.
- [4] Hong M Y, Hajinezhad D, Zhao M M. Prox-PDA: The proximal primal-dual algorithm for fast distributed nonconvex optimization and learning over networks[C]. Proceedings of the 34th International Conference on Machine Learning. Sydney, 2017: 1529-1538.
- [5] Boyd S, Parikh N, Chu E, et al. Distributed optimization and statistical learning via the alternating direction method of multipliers[J]. Foundations and Trends[®] in Machine Learning, 2011, 3(1): 1-122.
- [6] Zheng Y L, Liu Q S. A review of distributed optimization: Problems, models and algorithms[J]. Neurocomputing, 2022, 483(1): 446-459.
- [7] Yang T, Yi X L, Wu J F, et al. A survey of distributed

- optimization[J]. *Annual Reviews in Control*, 2019, 47: 278-305.
- [8] 陈为胜. 分布式优化、学习理论与方法[M]. 北京: 科学出版社, 2019: 15-18.
(Chen W S. Theory and method of distributed optimization and learning[M]. Beijing: Science Press, 2019: 15-18.)
- [9] 衣鹏, 洪奕光. 分布式合作优化及其应用[J]. *中国科学: 数学*, 2016, 46(10): 1547-1564.
(Yi P, Hong Y G. Distributed cooperative optimization and its applications[J]. *Scientia Sinica: Mathematica*, 2016, 46(10): 1547-1564.)
- [10] 杨涛, 柴天佑. 分布式协同优化的研究现状与展望[J]. *中国科学: 技术科学*, 2020, 50(11): 1413-1425.
(Yang T, Chai T Y. Research status and prospects of distributed collaborative optimization[J]. *Scientia Sinica: Technologica*, 2020, 50(11): 1413-1425.)
- [11] Nedić A, Ozdaglar A. Distributed subgradient methods for multi-agent optimization[J]. *IEEE Transactions on Automatic Control*, 2009, 54(1): 48-61.
- [12] Matei I, Baras J S. Performance evaluation of the consensus-based distributed subgradient method under random communication topologies[J]. *IEEE Journal of Selected Topics in Signal Processing*, 2011, 5(4): 754-771.
- [13] Shi W, Ling Q, Wu G, et al. EXTRA: An exact first-order algorithm for decentralized consensus optimization[J]. *SIAM Journal on Optimization*, 2015, 25(2): 944-966.
- [14] Qu G N, Li N. Harnessing smoothness to accelerate distributed optimization[J]. *IEEE Transactions on Control of Network Systems*, 2018, 5(3): 1245-1260.
- [15] Nedić A, Olshevsky A, Shi W. Achieving geometric convergence for distributed optimization over time-varying graphs[J]. *SIAM Journal on Optimization*, 2017, 27(4): 2597-2633.
- [16] Wang J, Elia N. Control approach to distributed optimization[C]. *Proceedings of the 48th Annual Allerton Conference on Communication, Control, and Computing*. Monticello, 2011: 557-561.
- [17] Kia S S, Cortés J, Martínez S. Distributed convex optimization via continuous-time coordination algorithms with discrete-time communication[J]. *Automatica*, 2015, 55: 254-264.
- [18] Yang S F, Liu Q S, Wang J. A multi-agent system with a proportional-integral protocol for distributed constrained optimization[J]. *IEEE Transactions on Automatic Control*, 2017, 62(7): 3461-3467.
- [19] 杨涛, 徐磊, 易新蕾, 等. 基于事件触发的分布式优化算法[J]. *自动化学报*, 2022, 48(1): 133-143.
(Yang T, Xu L, Yi X L, et al. Event-triggered distributed optimization algorithms[J]. *Acta Automatica Sinica*, 2022, 48(1): 133-143.)
- [20] Wang X Y, Wang G D, Li S H. Distributed finite-time optimization for integrator chain multiagent systems with disturbances[J]. *IEEE Transactions on Automatic Control*, 2020, 65(12): 5296-5311.
- [21] Shi X S, Xu L, Yang T, et al. Distributed fixed-time resource allocation algorithm for the general linear multi-agent systems[J]. *IEEE Transactions on Circuits and Systems II: Express Briefs*, 2022, 69(6): 2867-2871.
- [22] Chen G, Ren J H, Feng E N. Distributed finite-time economic dispatch of a network of energy resources[J]. *IEEE Transactions on Smart Grid*, 2017, 8(2): 822-832.
- [23] Yao L S, Yuan Y, Sundaram S, et al. Distributed finite-time optimization[C]. *Proceedings of the 14th International Conference on Control and Automation*. Anchorage, 2018: 147-154.
- [24] Li C J, Yu X H, Zhou X J, et al. A fixed time distributed optimization: A sliding mode perspective[C]. *Proceedings of the 43rd Annual Conference of the IEEE Industrial Electronics Society*. Beijing, 2017: 8201-8207.
- [25] Ning B D, Han Q L, Zuo Z Y. Practical fixed-time consensus for integrator-type multi-agent systems: A time base generator approach[J]. *Automatica*, 2019, 105: 406-414.
- [26] Guo Z J, Chen G. Predefined-time distributed optimal allocation of resources: A time-base generator scheme[J]. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 2022, 52(1): 438-447.
- [27] Colunga J A, Vázquez C R, Becerra H M, et al. Predefined-time consensus of nonlinear first-order systems using a time base generator[J]. *Mathematical Problems in Engineering*, DOI: 10.1155/2018/1957070.
- [28] Li S L, Nian X H, Deng Z H, et al. Predefined-time distributed optimization of general linear multi-agent systems[J]. *Information Sciences*, 2022, 584: 111-125.
- [29] Liu Y, Xia Z C, Gui W H. Multi-objective distributed optimization via A predefined-time multi-agent approach[J]. *IEEE Transactions on Automatic Control*, DOI: 10.1109/TAC.2023.3244122.
- [30] Zhang K P, Xu L, Yi X L, et al. Predefined-time distributed multi-objective optimization for network resource allocation[J]. *Science China Information Sciences*, DOI: 10.1007/s11432-022-3791-8.
- [31] Agarwal A, Negahban S, Wainwright M J. Fast global convergence of gradient methods for high-dimensional statistical recovery[J]. *The Annals of Statistics*, 2012, 40(5): 2452-2482.
- [32] Yi X L, Yao L S, Yang T, et al. Distributed optimization for second-order multi-agent systems

- with dynamic event-triggered communication[J/OL]. 2018, arXiv: 1803.06380.
- [33] Lai M J, Yin W T. Augmented ℓ_1 and nuclear-norm models with a globally linearly convergent algorithm[J]. SIAM Journal on Imaging Sciences, 2013, 6(2): 1059-1091.
- [34] Garg K, Baranwal M, Panagou D. A fixed-time convergent distributed algorithm for strongly convex functions in a time-varying network[C]. Proceedings of the 59th IEEE Conference on Decision and Control. Jeju, 2021: 4405-4410.
- [35] Chen G, Li Z Y. A fixed-time convergent algorithm for distributed convex optimization in multi-agent systems[J]. Automatica, 2018, 95: 539-543.
- [36] 张苗苗, 叶茂娇, 郑元世. 预设时间下的分布式优化和纳什均衡点求解[J]. 控制理论与应用, 2022, 39(8): 1397-1406.
(Zhang M M, Ye M J, Zheng Y S. Prescribed-time distributed optimization and Nash equilibrium seeking[J]. Control Theory & Applications, 2022, 39(8): 1397-1406.)
- [37] Godsil C D, Royle G. Algebraic graph theory[M]. New York: Springer, 2001: 207.
- [38] Becerra H M, Vázquez C R, Arechavaleta G, et al. Predefined-time convergence control for high-order integrator systems using time base generators[J]. IEEE Transactions on Control Systems Technology, 2018, 26(5): 1866-1873.
- [39] 时侠圣, 孙佳月, 徐磊, 等. 二阶智能体的分布式非光滑资源分配算法[J]. 控制与决策, 2023, 38(5): 1336-1344.
(Shi X S, Sun J Y, Xu L, et al. Distributed nonsmooth resource allocation algorithm over second-order agents[J]. Control and Decision, 2023, 38(5): 1336-1344.)

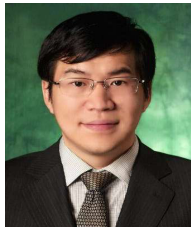
作者简介

杨涛(1981—),男,教授,博士生导师,从事信息物理系统、复杂工业过程的分布式优化等研究,E-mail: yangtao@mail.neu.edu.cn;

常怡然(1999—),女,硕士生,从事分布式优化、混合整数优化等研究,E-mail: 2102019@stu.neu.edu.cn;

张坤朋(1995—),男,博士生,从事分布式优化、在线优化等研究,E-mail: 2110343@stu.neu.edu.cn;

徐磊(1994—),男,博士生,从事分布式非凸优化等研究,E-mail: 2010345@stu.neu.edu.cn.



特邀专家 杨涛,东北大学教授、博士生导师. 2019年入选国家青年高层次人才类项目,加入东北大学流程工业综合自动化国家重点实验室. 主要从事工业人工智能、智能优化与控制一体化、信息物理系统、分布式协同控制和优化等领域的研究. 主持国家自然科学基金重点项目、重大项目课题、国家重点研发计划课题等. 研究成果发表论文百余篇,其中IEEE汇刊和IFAC会刊论文30余篇. 2018年获美国橡树岭大学联盟Ralph E Powe青年教授奖;多次获国际会议最佳论文奖、最佳学生论文奖. 现任《自动化学报》副主编、IEEE TCST、IEEE TNNLS等期刊编委. 担任中国自动化学会大数据专委会主任委员、中国人工智能学会工业人工智能常务副秘书长等.

专家寄语 很荣幸能见证东北大学百年校庆这一历史时刻,首先祝东北大学百岁生日快乐,继往开来,培养出更多自强不息,知行合一的优秀人才!同时祝东北大学控制学科在服务国家重大需求和东北振兴中发挥引领作用,再创辉煌!