KYBERNETIKA — VOLUME *61* (2025), NUMBER 3, PAGES 377–403

ADAPTIVE FRACTIONAL DISTRIBUTED OPTIMIZATION ALGORITHM WITH DIRECTED SPANNING TREES

Huaijin Peng, Yiheng Wei, Shuaiyu Zhou, and DongDong Yue

Distributed optimization has garnered significant attention in past decade, yet existing algorithms mainly rely on Laplacian matrix information for parameter settings, limiting their adaptability and applicability. To design the fully distributed algorithm, this paper uses an adaptive weight framework based on directed spanning trees (DST), which not only solves the consensus optimization problem but also can be extended to solve the resource allocation problem. The innovative integration of Nabla fractional calculus further improves performance, enabling efficient discrete-time distributed optimization. Moreover, The proposed algorithms optimality and convergence properties have been rigorously analyzed, which demonstrates that they can converge to the optimal solution of the problem under consideration. Finally, numerical simulations are conducted to validate the algorithm's feasibility and superiority.

Keywords: distribute optimization, fractional calculus, directed graphs, directed spanning trees, resource allocation, fully distributed

Classification: 05C05,05C20,26A33,90C26

1. INTRODUCTION

In past decade, with the development across multiple research fields, distributed optimization has garnered significant attention and extensive study. Unlike centralized optimization, distributed optimization operates by allowing multiple agents to work together, each solving a part of the problem and sharing information with others in order to obtain the optimal solution. Distributed optimization has achieved in various applications such as autonomous driving, smart grids and distributed computing [1, 4, 6, 9, 16, 18, 27, 40].

To address distributed optimization, algorithm design must consider the network's communication topology, such as undirected connected graphs [14, 15, 17, 25, 28], and digraphs [4, 7, 11, 12, 20, 24, 33] and so on. These algorithms in [7, 11, 12, 20, 33] typically rely on the Laplacian matrix's eigenvalues or eigenvectors for parameter setting. An algorithm in [24] introduces saddle points for not requiring the knowledge of Laplacian matrix but faces challenges with vanishing step sizes. Meanwhile, all aforementioned algorithms except [12] require convexity of the local functions. When the network is large and sparse, the strategy of setting parameters based on the global

DOI: 10.14736/kyb-2025-3-0377

Laplacian matrix with a static gain may lead to high gain and instability. The above algorithms require global Laplace matrix information or global Lipschitz continuity and are not fully distributed algorithms. Recently, one possible solution is to consider fully distributed algorithms. The fully distributed algorithm does not need to rely on the global Lipschitz continuity of the gradient, nor does it require a priori global Laplacian information [35, 38, 39]. The fully distributed adaptive algorithm with DST in [35, 38] is proposed to address the consensus optimization problem of the Lagrange multipliers. A DST can be found in a distributed way without any knowledge of the Laplacian matrix [10]. However, algorithms in [35, 38] only achieve asymptotic convergence.

In additon, fractional calculus, as an extension of classical calculus, allows derivatives and integrals to have arbitrary real orders. In recent years, with the deepening study of complex systems, fractional calculus has gained increasing attention and has been widely applied in various fields [3, 13, 21, 29, 30]. The algorithm in [13] shows superior performance, achieving a convergence speed that surpasses that of the integer order algorithm. Although fractional calculus has been proven to deliver significant performance improvements, its application in distributed optimization algorithms is still in its early stages [5, 23, 26, 34]. The study in [34] investigates the distributed optimization problem for fractional nonlinear uncertain multi-agent systems with unmeasured states. The algorithm in [26] solves the nonlinear fractional fixed-time distributed time-varying optimization problem over unbalanced directed graphs. However, most of existing algorithms are continuous-time algorithms, which require real-time communication and gradient computation, leading to increased communication and computational costs. It is worth mentioning that we have introduced Nabla discrete-time fractional calculus into the distributed optimization algorithm, significantly reducing computational and communication cost [8, 19, 37, 41]. Similarly, the aforementioned distributed algorithms face the issue of relying on global Laplacian matrix information.

This work explores fractional distributed optimization on directed graphs. It introduces fractional calculus and design the DST adaptive gain framework. Thus enable agents to self-determine edge coupling strength based on DST, promoting consensus on the Lagrangian multiplier of the optimal solution. It separately investigates the DST-based fractional distributed optimization algorithm and fractional distributed resource allocation algorithm. The advantages of these algorithms are as follows,

i) This paper designs DST-based adaptive fractional distributed optimization algorithm and resource allocation algorithm, which is applied to solve the distributed optimization consensus problem and distributed resource allocation problem over directed graphs, with both proving their Mittag–Leffler convergence.

ii) Unlike traditional distributed algorithms, this paper designs the fully distributed algorithm. Influenced by the viewpoint of uncertain saddle-point dynamics, this paper designs an adaptive coupling gain framework based on DST, removing the dependency on global Laplacian matrix information, eliminating the need for vanishing step sizes, and in optimization problems, relaxing the convexity requirement for local cost functions, significantly enhancing flexibility and applicability. In the DST-based adaptive strategy, only the gains along the edges associated with the DST are adaptive.

iii) This paper combined fractional calculus with adaptive weight mechanisms, the incorporation of fractional calculus, particularly Nabla fractional calculus, enriches the algorithm's dynamic characteristics, allowing it to better capture memory and hereditary properties, improves algorithms performance. And it enables the transition from continuous-time to discrete-time implementations, ensuring robustness and feasibility.

The rest of the paper is organized as follows. Section 2 introduces the preamble and the problem setup. Section 3 describes DST-based fractional distributed optimization algorithm and fractional distributed resource allocation and their Mittag–Leffler convergence is obtained through proofs. Section 4 verifies the performance of the algorithms numerically through simulations. Section 5 concludes and discusses some future directions.

2. PRELIMINARIES

2.1. Notation

The real coordinate space with appropriate dimensions is denoted by $\mathbb{R}$ and $\mathbb{R}_+$ is the real positive scalar subspace. $\mathbb{Z}_+$ usually represents the set of positive integers. $\mathbb{N}_a$ represents the set $\{a,\ a+1,\ \dots\}$, where $a \in \mathbb{R}$. $\mathcal{I}_N$ represents the set $\{1,\ \dots,\ N\}$. Define the N-dimensional identity matrix by $\mathbf{I}_N$ and 1_N, and the column vector with N elements being one. 0 denotes a column vector with all zeros. $\mathcal{M}_{\mathrm{r}}^N$ is a set of $n \times n$ matrices with zero row sums. Denote $\mathrm{col}(x_1,\ \dots,\ x_N) = [x_1^\top,\ \dots,\ x_N^\top]^\top$ as the column vectors. $\otimes$ is the Kronecker product. $*$ indicates the convolution operation, i. e., $x(k) * y(k) = \sum_{j=a+1}^{k} x(k-j+a+1)y(j).$, for $x, y : \mathbb{N}_{a+1} \to \mathbb{R}$, $a \in \mathbb{R}$,. $\binom{p}{q} = \frac{\Gamma(p+1)}{\Gamma(q+1)\Gamma(p-q+1)}$, and $\Gamma(x) = \int_0^{+\infty} \mathrm{e}^{-t} t^{x-1}\,\mathrm{d}t$. Let matrix $A^\top$ be the transpose of A. And denote $A^{\mathrm{s}} = (A + A^\top)/2$ as the undirected version of A. Denote $\bar{\lambda}$ (or $\underline{\lambda}$) as the maximum (or minimum) eigenvalue of the symmetric matrix A. Denote the gradient of the f differentiable function by ∇f. If a contineously differentiable funtion $f : \mathbb{R}^N \to \mathbb{R}$ is strictly convex, then there is a convex set Ω if $(x-y)^\top[\nabla f(x) - \nabla f(y)] > 0$, $\forall x,\, y \in \Omega$ with $x \neq y$. Define the value of the discrete optimization variable x at time k by $x(k)$, and k is omitted after. ${}_a^{\mathrm{C}}\nabla_k^{\alpha} x(k)$ denotes the derivative of the α-order nabla discrete fractional with respect to k under the definition of Caputo. The function $p^{\overline{q}} = \frac{\Gamma(p+q)}{\Gamma(p)}$ is called the rising function, where $p \in \mathbb{R}$ and $q \in \mathbb{R}$.

From the asymptotic properties of the Gamma function, we have $\lim_{p\to+\infty} \frac{p^{\overline{q}}}{p^q} = 1$. Using mathematical induction and the definition of the rising function, and referring to [3−5], the following basic properties can be easily derived:

2.2. Graph Theory

A weighted directed graph [22] $\mathcal{G}(\mathcal{V}, \mathcal{E}, \mathcal{W})$ consists of a node set $\mathcal{V} = \mathcal{I}_N$, an edge set $\mathcal{E} = \{e_{ij} \mid i \neq j; i \to j\}$, and a weighted adjacency matrix $\mathcal{W} = (w_{ij}) \in \mathbb{R}^{N\times N}$. If $e_{ij} \in \mathcal{E}$, then i is termed an in-neighbor of j, and the set of all in-neighbors of j is denoted as $\mathcal{N}_{\mathrm{in}}(j)$. Similarly, $\mathcal{N}_{\mathrm{out}}(i)$ represents the set of all out-neighbors of i.

The Laplacian matrix $\mathcal{L} = (\mathcal{L}_{ij}) \in \mathbb{R}^{N\times N}$ of $\mathcal{G}$ is defined such that $\mathcal{L}_{ij} = -w_{ij}$ for $i \neq j$ and $\mathcal{L}_{ii} = \sum_k w_{ik}$ for $i = 1, \ldots, N$. A path refers to a sequence of edges that connects a pair of nodes. A directed graph $\mathcal{G}$ is considered strongly connected if any pair of nodes is connected by a directed path and weakly connected if any pair of nodes is connected by a path ignoring the direction of edges. Moreover, the graph is weight-balanced if $\sum_{j\in\mathcal{N}_{\text{in}}(i)} w_{ij} = \sum_{j\in\mathcal{N}_{\text{out}}(i)} w_{ji}$ for all $i \in \mathcal{V}$.

A DST $\bar{\mathcal{G}}(\mathcal{V}, \bar{\mathcal{E}}, \bar{\mathcal{W}})$ is a spanning tree that originates from a root node with no in-neighbors and can reach each subsequent node along directed edges, with each node having a unique in-neighbor except for the root. Let i_p be the unique in-neighbor of node $p+1$ in $\bar{\mathcal{G}}$. Correspondingly, $\bar{\mathcal{L}}$ and $\bar{\mathcal{W}}$ are the Laplacian matrix and the weighted adjacency matrix of $\bar{\mathcal{G}}$, respectively. The set of out-neighbors of i in $\bar{\mathcal{G}}$ is denoted as $\bar{\mathcal{N}}_{\text{out}}(i)$.

2.3. Nabla fractional calculus

Definition 2.1. (Wei et al. [32]) The αth Grünwald–Letnikov fractional sum of function $f : \mathbb{N}_{a+1} \to \mathbb{R}$ is defined as

$$ {}_a^{\mathrm{G}}\nabla_k^{-\alpha} f(k) := \sum\nolimits_{i=0}^{k-a-1} (-1)^i \binom{-\alpha}{i} f(k-i), \tag{1} $$

where $\alpha > 0$, $k \in \mathbb{N}_{a+1}$, $a \in \mathbb{R}$.

Definition 2.2. (Wei et al. [32]) The αth Caputo fractional difference of function $f : \mathbb{N}_{a+1-n} \to \mathbb{R}$ is defined as

$$ {}_a^{\mathrm{C}}\nabla_k^{\alpha} f(k) := {}_a^{\mathrm{G}}\nabla_k^{\alpha-n} \nabla^n f(k), \tag{2} $$

where $\alpha \in (n-1, n), n \in \mathbb{Z}_+$, $k \in \mathbb{N}_{a+1}$, $a \in \mathbb{R}$.

Lemma 2.3. (Wei et al. [32]) For any $\alpha \in (0,1)$, $y(k) \in \mathbb{R}^n$, $n \in \mathbb{Z}_+$, $k \in \mathbb{N}_{a+1}$, $a \in \mathbb{R}$ and the positive definite matrix $P \in \mathbb{R}^{n\times n}$, it has the following inequality

$$ {}_a^{\mathrm{C}}\nabla_k^{\alpha} y^\top(k) P y(k) \leq 2y^\top(k) P {}_a^{\mathrm{C}}\nabla_k^{\alpha} y(k). \tag{3} $$

Definition 2.4. (Wei et all. [31]) The discrete-time Mittag–Leffler function based on the time domain is defined as

$$ \mathcal{F}_{\alpha,\beta}(\lambda, k, a) := \sum_{i=0}^{+\infty} \frac{\lambda^i (k-a)^{\overline{i\alpha+\beta-1}}}{\Gamma(i\alpha+\beta)}, \tag{4} $$

where $\alpha > 0$, $\beta > 0$, $\lambda \in \mathbb{C}$, $k \in \mathbb{N}_{a+1}$, and $a \in \mathbb{R}$. By taking the inverse nabla Laplace transform, it can be expressed as follows,

$$ \mathcal{F}_{\alpha,\beta}(\lambda, k, a) := \mathcal{N}_a^{-1}\left\{ \frac{s^{\alpha-\beta}}{s^\alpha - \lambda} \right\}, \tag{5} $$

where $s \in \mathbb{C}$, and $\mathcal{N}_a^{-1}\{\cdot\}$ denotes the inverse nabla Laplace transform.

2.4. Technical Lemmas

Lemma 2.5. (Yue et al. [36]) Suppose $\mathcal{G}$ contains a DST $\bar{\mathcal{G}}$. Let $\tilde{\mathcal{L}} = \mathcal{L} - \bar{\mathcal{L}}$.

Define $\Xi = (\Xi_{pj}) \in \mathbb{R}^{(N-1)\times N}$ to determine the edge relationship between p and j in $\bar{\mathcal{L}}$ as

$$\Xi_{pj} = \begin{cases} -1, & \text{if } j = p+1, \\ 1, & \text{if } j = i_p, \\ 0, & \text{otherwise.} \end{cases} \tag{6}$$

Define $Q = (Q_{pj}) \in \mathbb{R}^{(N-1)\times(N-1)} := \tilde{Q} + \bar{Q}$ with

$$\begin{aligned} Q_{pj} =& \tilde{Q}_{pj} + \bar{Q}_{pj}, \\ \tilde{Q}_{pj} =& \sum\nolimits_{c\in\bar{\mathcal{V}}_{j+1}} (\tilde{\mathcal{L}}_{p+1,c} - \tilde{\mathcal{L}}_{i_p,c}), \\ \bar{Q}_{pj} =& \sum\nolimits_{c\in\bar{\mathcal{V}}_{j+1}} (\bar{\mathcal{L}}_{p+1,c} - \bar{\mathcal{L}}_{i_p,c}), \end{aligned} \tag{7}$$

where $\bar{\mathcal{V}}_{j+1}$ represents the vertex set of the subtree of $\bar{\mathcal{G}}$ rooting at node $j+1$. Then, the following statements hold

1. $\mathcal{L}$ has a simple zero eigenvalue corresponding to the right eigenvector 1_N, and the other eigenvalues have positive real parts.

2. $\Xi\mathcal{L} = Q\Xi$.

3. $\bar{Q}$ can be explicitly written as

$$\bar{Q}_{pj} = \begin{cases} \bar{w}_{j+1,i_j}, & \text{if } j = p, \\ -\bar{w}_{j+1,i_j}, & \text{if } j = i_p - 1, \\ 0, & \text{otherwise.} \end{cases}$$

4. The eigenvalues of Q are exactly the nonzero eigenvalues of $\mathcal{L}$.

Lemma 2.6. (Bullo et al. [2]) A binary alphabet $\mathcal{G}$ with N nodes is weight-balanced iff $1_N^\top \mathcal{L} = 0$.

Lemma 2.7. Consider the Lyapunov function

$$V = \frac{1}{2}\left(X^\top X + Y^\top Y\right), \tag{8}$$

where X and Y are variables. Assume that there exists a constant $\mu > 0$ such that

$${}_a^{\mathrm{C}}\nabla_k^\alpha V \le -\mu\left(\frac{1}{2}X^\top X\right). \tag{9}$$

Then, it holds that X converges to 0 with the Mittag–Leffler rate.

Proof. From (9), the fractional sum is applied simultaneously to both sides of the inequality, and it can be obtains that

$$ {}_{a}^{\mathrm{G}}\nabla_{k}^{-\alpha}\,{}_{a}^{\mathrm{C}}\nabla_{k}^{\alpha}V = V(k) - V(a) \le -\frac{1}{2}\mu\,{}_{a}^{\mathrm{G}}\nabla_{k}^{-\alpha}X^{\top}X, \tag{10} $$

which implys that

$$ V(k) \le V(a) - \frac{1}{2}\mu\,{}_{a}^{\mathrm{G}}\nabla_{k}^{-\alpha}X^{\top}X. \tag{11} $$

It is inferred from (8) and (11) that

$$ \begin{aligned} \frac{1}{2}X^{\top}X &\le \frac{1}{2}\left(X^{\top}X + Y^{\top}Y\right) \\ &\le \frac{1}{2}\left(X(a)^{\top}X(a) + Y(a)^{\top}Y(a)\right) - \frac{1}{2}\mu\,{}_{a}^{\mathrm{G}}\nabla_{k}^{-\alpha}X^{\top}X. \end{aligned} \tag{12} $$

By defining $x(k) = \frac{1}{2}X^{\top}X$, and $\lambda = \frac{1}{2}[X(a)^{\top}X(a) + Y(a)^{\top}Y(a)]$, (12) reads

$$ x(k) \le \lambda - \mu\,{}_{a}^{\mathrm{G}}\nabla_{k}^{-\alpha}x(k). \tag{13} $$

Denote $m(k) = \lambda - \mu\,{}_{a}^{\mathrm{G}}\nabla_{k}^{-\alpha}x(k) - x(k) \ge 0$, for every $k \in \mathbb{N}_{a+1}$, one has

$$ x(k) + m(k) = \lambda - \mu\,{}_{a}^{\mathrm{G}}\nabla_{k}^{-\alpha}x(k). \tag{14} $$

By taking the nabla Laplace transform on both side of (14), it yields that

$$ x_f(s) + m_f(s) = \frac{\lambda}{s} - \frac{\mu x_f(s)}{s^{\alpha}}, \tag{15} $$

where $x_f(s) = \mathcal{N}_a\{x(k)\}$, $m_f(s) = \mathcal{N}_a\{m(k)\}$. Then, it follows that

$$ x_f(s) = \frac{s^{\alpha-1}\lambda}{s^{\alpha}+\mu} - \frac{s^{\alpha}m_f(s)}{s^{\alpha}+\mu}. \tag{16} $$

Taking the inverse nabla Laplace transform on both sides, one has

$$ \begin{aligned} x(k) &= \lambda\mathcal{F}_{\alpha,1}(-\mu,k,a) - m(k) * \mathcal{F}_{\alpha,0}(-\mu,k,a) \\ &= \lambda\mathcal{F}_{\alpha,1}(-\mu,k,a) - m(k) * [1 - \mathcal{F}_{\alpha,\alpha}(-\mu,k,a)], \end{aligned} \tag{17} $$

Due to the properties of the Mittag–Leffler function, it holds that $\mathcal{F}_{\alpha,\alpha}(-\mu,k,a) \le 1$. Thus, one has

$$ x(k) \le \lambda\mathcal{F}_{\alpha,1}(-\mu,k,a). \tag{18} $$

Consequently, the convergence of $x(k)$ can be obtained. Due to $x(k) = \frac{1}{2}X^{\top}X$, X is Mittag–Leffler convergent. □

2.5. Problem setup

This work studies two critical problems: distributed consensus optimization and distributed resource allocation.

The problem of distributed consensus optimization has the following form as

$$\begin{aligned} \min_{x\in\mathbb{R}^{Nn}} \; & F(x) = \sum\nolimits_{i=1}^{N} f_i(x_i), \\ \text{s.t.} \quad & x_1 = x_2 = \cdots = x_N, \end{aligned} \tag{19}$$

where $x = \text{col}(x_1, \dots, x_N)$, $F(\cdot)$ is the global summation-separable cost function and $f_i(\cdot)$ is the local cost function for each agent. Consider N agents interacting over a digraph $\mathcal{G}(\mathcal{V}, \mathcal{E}, \mathcal{W})$, cooperatively seeking a global minimizer of (19), denoted by x^*.

To solve the above optimization problem in a distributed manner, the following assumptions are made.

Assumption 1. The global cost function $F(\cdot)$ is differentiable and strictly convex. Each local cost function $f_i(\cdot)$ is differentiable.

And for the distributed resource allocation problem, it is formulated as

$$\begin{aligned} \min_{x\in\mathbb{R}^{Nn}} \; & G(x) = \sum\nolimits_{i=1}^{N} g_i(x_i), \\ \text{s.t.} \quad & \sum\nolimits_{i=1}^{N} x_i = d, \end{aligned} \tag{20}$$

where $x = \text{col}(x_1, \dots, x_N)$, $d = \sum_{i=1}^{N} d_i$. Each agent has its local resources $d_i \in \mathbb{R}$ and is associated to a local cost function $g_i(\cdot) : \mathbb{R}^n \to \mathbb{R}$. Consider N agents communicating over a digraph $\mathcal{G}(\mathcal{V}, \mathcal{E}, \mathcal{W})$, cooperatively seeking a global allocation strategy with the minimum global cost function $G(\cdot)$ and satisfying the sum of the total resources.

For the distributed resource allocation, the following assumption is standard.

Assumption 2. The global cost function $G(\cdot)$ is differentiable and strictly convex. Each local cost function $g_i(\cdot)$ is differentiable.

Assumption 3. The digraph $\mathcal{G}$ is strongly connected and weight balanced.

3. MAIN RESULTS

3.1. Distributed consensus optimization problem

Under strongly connected graph conditions, DST can be obtained by a distributed method even without any prior information about the Laplacian matrix. So a DST-Based fractional distributed optimization algorithm with adaptive weights is proposed to solve problem (19) without such knowledge.

Consider any DST $\bar{\mathcal{G}}$ of $\mathcal{G}$ for the algorithm. Each agent $i \in \mathcal{V}$ has its own local estimate $x_i \in \mathbb{R}^n$ of the optimal decision variable x^* and the auxiliary variable $y_i \in \mathbb{R}^n$. Communication between agents is only through their in-neighbors. Each agent i communicates x_i over $\mathcal{G}^{\mathcal{B}}(\mathcal{V}, \mathcal{E}, \mathcal{B}(k))$, where $\mathcal{B}(k) = (b^{ij}(k))$ is the weight matrix for the dynamic coupling gain at $\mathcal{B}(a) = \mathcal{W}$ and communicates y_i over $\mathcal{G}$. Hereafter, k is omitted. Design the algorithm as follows,

$$
\begin{aligned}
{}_a^{\mathrm{C}}\nabla_k^{\alpha} x_i =& - \gamma_1 \nabla f_i(x_i) - \sum\nolimits_{j\in\mathcal{N}_{\mathrm{in}}(i)} b_{ij}(x_i - x_j) - \sum\nolimits_{j\in\mathcal{N}_{\mathrm{in}}(i)} w_{ij}(y_i - y_j), && \text{(21a)}\\
{}_a^{\mathrm{C}}\nabla_k^{\alpha} y_i =& \sum\nolimits_{j\in\mathcal{N}_{\mathrm{in}}(i)} b_{ij}(x_i - x_j), && \text{(21b)}
\end{aligned}
$$

with dynamic coupling gains

$$
b_{ij} = \begin{cases} w_{ij}, & \text{if } e_{ji} \in \mathcal{E}\backslash\bar{\mathcal{E}}, \\ \bar{b}_{p+1,i_p}, & \text{if } e_{ji} \in \bar{\mathcal{E}}, \end{cases} \tag{22a}
$$

$$
{}_a^{\mathrm{C}}\nabla_k^{\alpha} \bar{b}_{p+1,i_p} = \gamma_2 [(x_{i_p} - x_{p+1}) - \sum\nolimits_{j\in\bar{\mathcal{N}}_{\mathrm{out}}(p+1)} (x_{p+1} - x_j)]^{\top} (x_{i_p} - x_{p+1}), \tag{22b}
$$

for $\gamma_1, \gamma_2 \in \mathbb{R}_+$. From (22), each agent i and its in-neighbor j update only when communicating x_j and when the edge e_{ji} is in $\bar{\mathcal{G}}$. By defining $x = \mathrm{col}(x_1, \ldots, x_N)$, $y = \mathrm{col}(y_1, \ldots, y_N)$, $f(x) = \sum_{i=1}^{N} f_i(x_i)$, the algorithm (21) reads

$$
\begin{aligned}
{}_a^{\mathrm{C}}\nabla_k^{\alpha} x &= -\gamma_1 \nabla f(x) - (\mathcal{L}^{\mathcal{B}} \otimes \mathbf{I}_n) x - (\mathcal{L} \otimes \mathbf{I}_n) y, && \text{(23a)}\\
{}_a^{\mathrm{C}}\nabla_k^{\alpha} y &= (\mathcal{L}^{\mathcal{B}} \otimes \mathbf{I}_n) x. && \text{(23b)}
\end{aligned}
$$

Theorem 3.1. Suppose Assumptions 1 and 3 hold. If $(\bar{x}, \bar{y})$ is an equilibrium point of (23), then it holds that $\bar{x} = x^*$, where x^* is the global minimizer of (19).

Proof.

When $(\bar{x}, \bar{y})$ is an equilibrium of (23), thus

$$
\begin{aligned}
0 &= -\gamma_1 \nabla f(\bar{x}) - (\mathcal{L}^{\mathcal{B}} \otimes \mathbf{I}_n)\bar{x} - (\mathcal{L} \otimes \mathbf{I}_n)\bar{y}, && \text{(24a)}\\
0 &= (\mathcal{L}^{\mathcal{B}} \otimes \mathbf{I}_n)\bar{x}. && \text{(24b)}
\end{aligned}
$$

Let $x = 1_N \otimes x_0$, for some $x_0 \in \mathbb{R}^n$, so $(\mathcal{L}^{\mathcal{B}} \otimes \mathbf{I}_n)x = (\mathcal{L}^{\mathcal{B}} 1_N) \otimes (\mathbf{I}_n x_0)$. By Lemma 2.5, 1_N is the right eigenvector of $\mathcal{L}$ and $\mathcal{L}^{\mathcal{B}}$ corresponding to their simple zero eigenvalues, so it implies that $(\mathcal{L}^{\mathcal{B}} 1_N) \otimes (\mathbf{I}_n x_0) = 0$. Therefore, $\bar{x} = x = 1_N \otimes x_0$.

By Lemma 2.6, $1_N^{\top}\mathcal{L} = 0$. Therefore, left-multiplying (24a) by $1_N^{\top} \otimes \mathbf{I}_n$ results in

$$
\begin{aligned}
0 =& - \gamma_1 (1_N^{\top} \otimes \mathbf{I}_n) \nabla f(\bar{x}) - (1_N^{\top} \otimes I_n)(\mathcal{L}^{\mathcal{B}} \otimes \mathbf{I}_n)\bar{x} - (1_N^{\top} \otimes \mathbf{I}_n)(\mathcal{L}^{\mathcal{B}} \otimes \mathbf{I}_n)\bar{y} \\
=& - \gamma_1 (1_N^{\top} \otimes \mathbf{I}_n) \nabla f(\bar{x}),
\end{aligned} \tag{25}
$$

which implies that $\sum_{i=1}^{N} f_i(x_0) = 0$, i.e., $F(x_0) = 0$. According to the strict convexity of $F(\cdot)$, it leads to $x^* = 1_N \otimes x_0$. Hence, $\bar{x} = x^*$. □

Note that if $(\bar{x}, \bar{y})$ is an equilibrium point of (23), $(\bar{x},\ y + 1_N \otimes \kappa)$ will also be an equilibrium point of (23), for any $\kappa \in \mathbb{R}^n$. Let any equilibrium point $(\bar{x}, \bar{y})$ transferred to the origin with applying a change of coordinates

$$
\mu = x - \bar{x}, \tag{26a}
$$

$$\nu = y - \bar{y}, \tag{26b}$$
$$\bar{\mu} = (\Xi \otimes \mathbf{I}_n)\mu, \tag{26c}$$
$$\bar{\nu} = (\Xi \otimes \mathbf{I}_n)\nu, \tag{26d}$$

where $\bar{\mu} = \mathrm{col}(\bar{\mu}_1, \cdots, \bar{\mu}_{N-1})$ and $\bar{\mu}_p = \mu_{i_p} - \mu_{p+1}$, $p \in \mathcal{I}_{N-1}$. Consequently, by Lemma 2.5 and the properties of the Kronecker product, the algorithm (21) and the adaptive law (22) are as follows,

$$ {}_a^{\mathrm{C}}\nabla_k^{\alpha}\bar{\mu} = -\gamma_1(\Xi \otimes \mathbf{I}_n)h - (Q^{\mathcal{B}} \otimes \mathbf{I}_n)\bar{\mu} - (Q \otimes \mathbf{I}_n)\bar{\nu}, \tag{27a}$$
$$ {}_a^{\mathrm{C}}\nabla_k^{\alpha}\bar{\nu} = (Q^{\mathcal{B}} \otimes \mathbf{I}_n)\bar{\mu}, \tag{27b}$$
$$ {}_a^{\mathrm{C}}\nabla_k^{\alpha}\bar{b}_{p+1,i_p} = \gamma_2(\bar{\mu}_p - \textstyle\sum_{j\in\bar{\mathcal{N}}_{\mathrm{out}}(p+1)} \bar{\mu}_{j-1})^{\top}\bar{\mu}_p, \tag{27c}$$

where $h = \nabla f(\mu + \bar{x}) - \nabla f(\bar{x})$, and Q as well as $Q^{\mathcal{B}}$ are defined as in Lemma 2.5 based on the DST $\bar{\mathcal{G}}$. More specifically, $Q^{\mathcal{B}} = \tilde{Q} + \bar{Q}^{\mathcal{B}}$ contains the fixed matrix $\tilde{Q}$, and the following time-varying matrix

$$\bar{Q}_{pj}^{\mathcal{B}} = \begin{cases} \bar{b}_{j+1,i_j}, & \text{if } j = p, \\ -\bar{b}_{j+1,i_j}, & \text{if } j = i_p - 1, \\ 0, & \text{otherwise.} \end{cases} \tag{28}$$

Theorem 3.2. Under Assumptions 1 and 3, algorithm (21) and the adaptive law (22) drive $x = \mathrm{col}(x_1, \ldots, x_i, \ldots, x_N)$ to x^* with Mittag–Leffler rate for all $i \in \mathcal{V}$, and for any initial condition $x_i(a)$, $y_i(a) \in \mathbb{R}^n$. Moreover, the weights $\bar{b}_{p+1,i_p}$, $p \in \mathcal{I}_{N-1}$ converge to some finite constant value.

Proof. The first step is to prove for system (27) with arbitrary initial conditions, ($\bar{\mu}$, $\bar{\nu}$) Mittag–Leffler converges to the origin, and the weights $\bar{b}_{p+1,i_p}$, $p \in \mathcal{I}_{N-1}$, converge to some finite constant values.

Using the positive definiteness of the matrix Q^s, reflect the stability of (27) in the adaptive coupling weights $\bar{b}_{p+1,i_p}$, $p \in \mathcal{I}_{N-1}$. Consider the following Lyapunov function

$$V = V_{\mu} + V_{\nu}, \tag{29a}$$
$$V_{\mu} = \frac{1}{2}\bar{\mu}^{\top}\bar{\mu} + \sum_{p=1}^{N-1} \frac{1}{2\gamma_2}(\bar{b}_{p+1,i_p} - \phi_{p+1,i_p})^2, \tag{29b}$$
$$V_{\nu} = \frac{3\bar{\lambda}(Q^{\top}Q)}{\underline{\lambda}(Q^s)} \cdot \frac{1}{2}(\bar{\mu} + \bar{\nu})^{\top}(\bar{\mu} + \bar{\nu}), \tag{29c}$$

where $Q^s > 0$ is guaranteed by Lemma 2.5, and $\phi_{p+1,i_p} \in \mathbb{R}_+$, $p \in \mathcal{I}_{N-1}$ will be decided later. According to Lemma 2.3, the fractional difference of V_{μ} along the trajectory of (27) is

$$\begin{aligned} {}_a^{\mathrm{C}}\nabla_k^{\alpha}V_{\mu} \leq & -\gamma_1\bar{\mu}^{\top}(\Xi \otimes \mathbf{I}_n)h - \bar{\mu}^{\top}(Q^{\mathcal{B}} \otimes \mathbf{I}_n)\bar{\mu} - \bar{\mu}^{\top}(Q \otimes \mathbf{I}_n)\bar{\nu} \\ & + \textstyle\sum_{p=1}^{N-1}(\bar{b}_{p+1,i_p} - \phi_{p+1,i_p})(\bar{\mu}_p - \sum_{j+1\in\bar{\mathcal{N}}_{\mathrm{out}}(p+1)} \bar{\mu}_j)^{\top}\bar{\mu}_p. \end{aligned} \tag{30}$$

According to (28), it holds that when $j = p$, $\bar{b}_{p+1,i_p} = \bar{Q}^{\mathcal{B}}_{pp}$. When $j+1 \in \bar{\mathcal{N}}_{\text{out}}(p+1)$, it means that $i_j - 1 = p$, so $\bar{b}_{p+1,i_p} = -\bar{Q}^{\mathcal{B}}_{jp}$. Let $\Phi \in \mathbb{R}^{(N-1)\times(N-1)}$ is defined as

$$\Phi_{pj} = \begin{cases} \phi_{j+1,i_j}, & \text{if } j = p, \\ -\phi_{j+1,i_j}, & \text{if } j = i_p - 1, \\ 0, & \text{otherwise.} \end{cases} \tag{31}$$

Them, substituting (31) into (30) yields

$$\begin{aligned} &\textstyle\sum_{p=1}^{N-1}(\bar{b}_{p+1,i_p} - \phi_{p+1,i_p})(\bar{\mu}_p - \sum_{j+1\in\bar{\mathcal{N}}_{\text{out}}(p+1)}\bar{\mu}_j)^\top\bar{\mu}_p \\ &= \textstyle\sum_{p=1}^{N-1}(\bar{Q}^{\mathcal{B}}_{pp}\bar{\mu}_p + \sum_{j=1,j\neq p}^{N-1}\bar{Q}^{\mathcal{B}}_{jp}\bar{\mu}_j)^\top\bar{\mu}_p - (\Phi_{pp}\bar{\mu}_p + \sum_{j=1,j\neq p}^{N-1}\Phi_{jp}\bar{\mu}_j)^\top\bar{\mu}_p \\ &= \textstyle\sum_{p=1}^{N-1}\sum_{j=1}^{N-1}\bar{Q}^{\mathcal{B}}_{jp}\bar{\mu}_j^\top\bar{\mu}_p - \Phi_{jp}\bar{\mu}_j^\top\bar{\mu}_p \\ &= \bar{\mu}^\top[(\bar{Q}^{\mathcal{B}} - \Phi)\otimes\mathbf{I}_n]\bar{\mu}. \end{aligned} \tag{32}$$

From Assumption 1 and (30)-(32), it follows that

$$\begin{aligned} {}^{\mathrm{C}}_{a}\nabla^{\alpha}_{k}V_\mu \leq& -\gamma_1\bar{\mu}^\top(\Xi\otimes\mathbf{I}_n)h - \bar{\mu}^\top(Q^{\mathcal{B}}\otimes\mathbf{I}_n)\bar{\mu} - \bar{\mu}^\top(Q\otimes\mathbf{I}_n)\bar{\nu} \\ &+ \bar{\mu}^\top\left[(\bar{Q}^{\mathcal{B}} - \Phi)\otimes\mathbf{I}_n\right]\bar{\mu} \\ =& -\gamma_1\bar{\mu}^\top(\Xi\otimes\mathbf{I}_n)h - \bar{\mu}^\top\left[(Q^{\mathcal{B}} - \bar{Q}^{\mathcal{B}} + \Phi)\otimes\mathbf{I}_n\right]\bar{\mu} - \bar{\mu}^\top(Q\otimes\mathbf{I}_n)\bar{\nu} \\ \leq& -\gamma_1\underline{\lambda}(\Upsilon)\bar{\mu}^\top\bar{\mu} - \gamma_1\bar{\mu}^\top(\Xi\otimes\mathbf{I}_n)h' - \bar{\mu}^\top[(\tilde{Q} + \Phi)\otimes\mathbf{I}_n]\bar{\mu} - \bar{\mu}^\top(Q\otimes\mathbf{I}_n)\bar{\nu}, \end{aligned} \tag{33}$$

where $h' = \psi(\mu + \bar{x}) - \psi(\bar{x})$ with $\psi(x) = \text{col}(\psi_1(x_1), \cdots, \psi_N(x_N))$.

By using Young's inequality, it yields that

$$-\gamma_1\bar{\mu}^\top(\Xi\otimes\mathbf{I}_n)h' \leq \frac{\bar{\mu}^\top\bar{\mu}}{2} + \frac{\gamma_1^2 h'^\top(\Xi^\top\Xi\otimes\mathbf{I}_n)h'}{2} \leq \frac{\bar{\mu}^\top\bar{\mu}}{2} + \frac{\gamma_1^2\bar{\lambda}(\Xi^\top\Xi)h'^\top h'}{2}, \tag{34a}$$

$$-\bar{\mu}^\top(Q\otimes\mathbf{I}_n)\bar{\nu} \leq \frac{\bar{\mu}^\top\bar{\mu}}{2} + \frac{\bar{\nu}^\top(Q^\top Q\otimes\mathbf{I}_n)\bar{\nu}}{2} \leq \frac{\bar{\mu}^\top\bar{\mu}}{2} + \frac{\bar{\lambda}(Q^\top Q)\bar{\nu}^\top\bar{\nu}}{2}. \tag{34b}$$

Moreover, assume that $\|\psi(x)\| \leq \sqrt{N}K$ for all $x \in \mathbb{R}^{Nn}$, and K can be unknown. Hence, it holds as follows,

$$h'^\top h' \leq (\|\psi(\mu + x^*)\| + \|\psi(x^*)\|)^2 \leq 4NK^2. \tag{35}$$

From (33)-(35), it can be obtained that

$$\begin{aligned} {}^{\mathrm{C}}_{a}\nabla^{\alpha}_{k}V_\mu \leq& -\bar{\mu}^\top((\tilde{Q} + \Phi + \gamma_1\underline{\lambda}(\Upsilon)\mathbf{I}_{N-1})\otimes\mathbf{I}_n)\bar{\mu} + \bar{\mu}^\top\bar{\mu} \\ &+ \frac{\bar{\lambda}(Q^\top Q)}{2}\bar{\nu}^\top\bar{\nu} + 2NK^2\gamma_1^2\bar{\lambda}(\Xi^\top\Xi). \end{aligned} \tag{36}$$

By Lemma 2.3, the fractional difference of V_ν is computed as

$${}^{\mathrm{C}}_{a}\nabla^{\alpha}_{k}V_\nu \leq -\gamma_1\bar{\mu}^\top(\Xi\otimes\mathbf{I}_n)h - \bar{\mu}^\top(Q\otimes\mathbf{I}_n)\bar{\nu} - \gamma_1\bar{\nu}^\top(\Xi\otimes\mathbf{I}_n)h - \bar{\nu}^\top(Q\otimes\mathbf{I}_n)\bar{\nu}. \tag{37}$$

Like (34), by using Young's inequality and the positive definiteness of Q^s, it is inferred that

$$
\begin{aligned}
-\gamma_1\bar{\mu}^\top(\Xi\otimes\mathbf{I}_n)h \leq& \frac{\bar{\mu}^\top\bar{\mu}}{2}+2NK^2\gamma_1^2\bar{\lambda}(\Xi^\top\Xi), && \text{(38a)}\\
-\bar{\mu}^\top(Q\otimes\mathbf{I}_n)\bar{\nu} \leq& \frac{\bar{\lambda}(Q^\top Q)\bar{\mu}^\top\bar{\mu}}{\underline{\lambda}(Q^s)}+\frac{\underline{\lambda}(Q^s)\bar{\nu}^\top\bar{\nu}}{4}, && \text{(38b)}\\
-\gamma_1\bar{\nu}^\top(\Xi\otimes\mathbf{I}_n)h \leq& \frac{\underline{\lambda}(Q^s)\bar{\nu}^\top\bar{\nu}}{2}+\frac{2NK^2\gamma_1^2\bar{\lambda}(\Xi^\top\Xi)}{\underline{\lambda}(Q^s)}, && \text{(38c)}\\
-\bar{\nu}^\top(Q\otimes\mathbf{I}_n)\bar{\nu} \leq& -\underline{\lambda}(Q^s)\bar{\nu}^\top\bar{\nu}. && \text{(38d)}
\end{aligned}
$$

According to (37) and (38), one has

$$
{}_a^{\mathrm{C}}\nabla_k^\alpha V_\nu \leq \frac{\underline{\lambda}(Q^s)+2\bar{\lambda}(Q^\top Q)}{2\underline{\lambda}(Q^s)}\bar{\mu}^\top\bar{\mu}-\frac{\underline{\lambda}(Q^s)}{4}\bar{\nu}^\top\bar{\nu}+\frac{2NK^2\gamma_1^2\bar{\lambda}(\Xi^\top\Xi)\left[1+\underline{\lambda}(Q^s)\right]}{\underline{\lambda}(Q^s)}. \tag{39}
$$

According to (36) and (39), ${}_a^{\mathrm{C}}\nabla_k^\alpha V$ in (30) is upper bounded by

$$
{}_a^{\mathrm{C}}\nabla_k^\alpha V \leq -\bar{\mu}^\top[(\tilde{Q}+\Phi+\gamma_1\underline{\lambda}(\Upsilon)\mathbf{I}_{N-1})\otimes\mathbf{I}_n]\bar{\mu}-\frac{\bar{\lambda}(Q^\top Q)}{4}\bar{\nu}^\top\bar{\nu}+\eta_1\bar{\mu}^\top\bar{\mu}+\eta_2, \tag{40}
$$

where $\eta_1=1+\frac{3\bar{\lambda}(Q^\top Q)\underline{\lambda}(Q^s)+6\bar{\lambda}(Q^\top Q)^2}{2\underline{\lambda}(Q^s)^2}$ and $\eta_2=2NK^2\gamma_1^2\bar{\lambda}(\Xi^\top\Xi)\{1+\frac{3\bar{\lambda}(Q^\top Q)[1+\underline{\lambda}(Q^s)]}{\underline{\lambda}(Q^s)^2}\}$.

Let $\delta\in\mathbb{R}_+$ be an arbitrarily small positive scalar. If $\bar{\mu}^\top\bar{\mu}\geq\delta$, then for any $\eta_1,\eta_2\in\mathbb{R}_+$, there exists a sufficiently large $\eta\in\mathbb{R}_+$ satisfying $\eta\geq\eta_1+\frac{\eta_2}{\bar{\mu}^\top\bar{\mu}}$, ensuring that $\eta\bar{\mu}^\top\bar{\mu}\geq\eta_1\bar{\mu}^\top\bar{\mu}+\eta_2$. Thus, the inequality holds

$$
{}_a^{\mathrm{C}}\nabla_k^\alpha V \leq -\bar{\mu}^\top\{[\Phi^s-\eta\mathbf{I}_{N-1}+\tilde{Q}^s+\gamma_1\underline{\lambda}(\Upsilon)\mathbf{I}_{N-1}]\otimes\mathbf{I}_n\}\bar{\mu}-\frac{\bar{\lambda}(Q^\top Q)}{4}\bar{\nu}^\top\bar{\nu}, \tag{41}
$$

where

$$
\Phi^s=\begin{bmatrix}
\phi_{2,i_1} & \frac{1}{2}\phi_{21} & \cdots & \frac{1}{2}\phi_{N-2,1} & \frac{1}{2}\phi_{N-1,1}\\
\frac{1}{2}\phi_{21} & \phi_{3,i_2} & \cdots & \cdots & \frac{1}{2}\phi_{N-1,2}\\
\vdots & \vdots & \ddots & \vdots & \vdots\\
\frac{1}{2}\phi_{N-2,1} & \vdots & \cdots & \phi_{N-1,i_{N-2}} & \frac{1}{2}\phi_{N-1,N-2}\\
\frac{1}{2}\phi_{N-1,1} & \frac{1}{2}\phi_{N-1,2} & \cdots & \frac{1}{2}\phi_{N-1,N-2} & \phi_{N,i_{N-1}}
\end{bmatrix}.
$$

The aim is to ensure ${}_a^{\mathrm{C}}\nabla_k^\alpha V\leq 0$. From (41), clearly it holds true when $\Phi^s-\eta\mathbf{I}_{N-1}>0$, where Φ^s is decided by the choice of appropriate ϕ_{p+1,i_p}. Denote $\Omega_1=[\phi_{2,i_1}-\eta]$, and $\Omega_k=\begin{bmatrix}\Omega_{p-1} & \varphi_p\\ \varphi_p^\top & \phi_{p+1,i_p}-\eta\end{bmatrix}$, where $\varphi_p=\frac{1}{2}[\phi_{p1},\,\phi_{p2},\,\cdots,\,\phi_{p,p-1}]^\top$, $p=2,\,\cdots,\,N-1$. When $\phi_{2,i_1}>\eta$, $\Omega_1>0$. Assume $\Omega_{p-1}>0$, $p\geq 2$. Due to $|\phi_{pj}|\leq\left|\phi_{j+1,i_j}\right|,\forall j\in\mathcal{I}_{p-1}$ in (34), $\varphi_p^\top\Omega_{p-1}^{-1}\varphi_p\leq\frac{\sum_{j=2}^p\phi_{j,i_{j-1}}^2}{4\underline{\lambda}(\Omega_{p-1})}$. So when choosing $\phi_{p+1,i_p}>\eta+\frac{\sum_{j=2}^p\phi_{j,i_{j-1}}^2}{4\underline{\lambda}(\Omega_{p-1})}$, $\Omega_p>0$. Through mathematical induction, it follows that $\Phi^s-\eta\mathbf{I}_{N-1}=\Omega_{N-1}$ is positive definite.

Moreover, since $\tilde{Q}$ and Υ are fixed, choosing sufficiently large ϕ_{p+1,i_p} always ensures that $\underline{\lambda}(\Phi^s - \eta\mathbf{I}_{N-1} + \tilde{Q}^s) > -\gamma_1\underline{\lambda}(\Upsilon)$. Then, $[\Phi^s - \eta\mathbf{I}_{N-1} + \tilde{Q}^s + \gamma_1\underline{\lambda}(\Upsilon)\mathbf{I}_{N-1}]$ is positive definite. Let $M = [\Phi^s - \eta\mathbf{I}_{N-1} + \tilde{Q}^s + \gamma_1\underline{\lambda}(\Upsilon)\mathbf{I}_{N-1}]$. From (41), it can be obtained that

$$ {}_{a}^{\mathrm{C}}\nabla_k^\alpha V \le -2\underline{\lambda}(M)(\frac{1}{2}\bar{\mu}^\top\bar{\mu}) - \frac{\bar{\lambda}(Q^\top Q)}{2}(\frac{1}{2}\bar{\nu}^\top\bar{\nu}) \tag{42} $$

Obviously, it follows that ${}_{a}^{\mathrm{C}}\nabla_k^\alpha V \le -2\underline{\lambda}(M)(\frac{1}{2}\bar{\mu}^\top\bar{\mu})$. Then, according to (29) and Lemma 2.7, it is concluded that $\bar{\mu}$ converges to the origin with Mittag–Leffler rate. Similarly, (42) can be computed as ${}_{a}^{\mathrm{C}}\nabla_k^\alpha V \le -\frac{\bar{\lambda}(Q^\top Q)}{2}(\frac{1}{2}\bar{\nu}^\top\bar{\nu})$. Therefore, based on the Mittag–Leffler convergence of $\bar{\mu}$ and Lemma 2.7, $\bar{\nu}$ is also Mittag–Leffler convergent.

Thus $(\bar{\mu}, \bar{\nu})$ converges to the origin with Mittag–Leffler rate and the weights $\bar{b}_{p+1,i_p}$, $p \in \mathcal{I}_{N-1}$, converge to some finite constant values.

The next step is to prove the algorithm (21) and the adaptive law (22) drive $x = \mathrm{col}(x_1, \ldots, x_i, \ldots, x_N)$ to x^* with Mittag–Leffler rate for all $i \in \mathcal{V}$. Since $(\bar{\mu}, \bar{\nu})$ converges to the origin with Mittag–Leffler rate and null-space of Ξ is spanned by 1_N, (x, y) in (23) also converges to $(\bar{x} + 1_N \otimes \tau,\ \bar{y} + 1_N \otimes \kappa)$ for some $\tau,\ \kappa \in \mathbb{R}^n$, in the original coordinates. According to the uniqueness of the optimizer x^*, seek a contradiction to find $\tau = 0$.

Then, assume $\tau \neq 0$ and the steady-state dynamics of τ can be obtained by (23) as

$$ \begin{aligned} 0 = {}_{a}^{\mathrm{C}}\nabla_k^\alpha \tau =& \frac{1}{N}(1_N^\top \otimes I_N)\,{}_{a}^{\mathrm{C}}\nabla_k^\alpha x \\ =& -\frac{\gamma_1}{N}\nabla F(x^* + \tau) - \frac{1}{N}(\mathbf{1}_N^\top \mathcal{L}^{\mathcal{B}}\mathbf{1}_N \otimes (x^* + \tau)) \\ &- \frac{1}{N}(\mathbf{1}_N^\top \mathcal{L} \otimes \mathbf{I}_n)(\bar{y} + \mathbf{1}_N \otimes \kappa) \\ =& -\frac{\gamma_1}{N}\nabla F(x^* + \tau) \neq 0, \end{aligned} \tag{43} $$

which is a contradiction. Thus $\tau = 0$. Therefore, any trajectory of (23) converges to an equilibrium point $(\bar{x},\ \bar{y} + 1_N \otimes \kappa)$, for some $\kappa \in \mathbb{R}^n$. According to Theorem 3.1, the agents' estimates $\mathrm{col}(x_1, \ldots, x_i, \ldots, x_N)$ converge to the optimizer x^* of (19). □

In this subsection, an adapted fractional distributed optimization algorithm is proposed. It relies on DST to design a new adapted framework, which means that dynamic coupling gains update as the agents communicate along the DST $\bar{G}$. Theorem 3.1 and Theorem 3.2 respectively prove the optimality and convergence of the algorithm. The former analyzes the relationship between the equilibrium point and the optimal solution, while the latter shows that the algorithm can converge to the equilibrium point with Mittag–Leffler rate from any initial value so that we can get the solution of problem (19).

The proposed adaptive DST fractional distributed optimization algorithm demonstrates significant advantages in solving distributed optimization problems:

i) The fully distributed algorithm eliminates the need for global Laplacian matrix information through an adaptive coupling weight mechanism, making it highly scalable for large-scale systems. The algorithm relaxes the convexity requirement for local cost functions, allowing non-convex objectives, which significantly expands its applicability to a wider range of real-world problems.

ii) The algorithm achieves significant improvements in convergence performance. Compared to the algorithm in [35], which only attains asymptotic convergence, the proposed algorithm realizes Mittag–Leffler convergence. Notably, when the fractional order $\alpha = 1$x, the proposed algorithm reduces to exponential convergence, not only maintaining a high convergence rate but also aligning with the convergence properties of classical integer algorithm.

iii) The integration of fractional calculus enhances the algorithm's dynamic properties, thereby improving convergence speed and optimization performance. Also, the fractional calculus enables the proposed algorithm to achieve discrete-time operation, making it more suitable for real-world applictions.

Remark 1. Because just using for proving convergence, K can be unknown. For any initial $x(a)$, $y(a) \in \mathbb{R}^{Nn}$, and any parameters γ_1, $\gamma_2 \in \mathbb{R}_+$, algorithm (21) can ensure convergence. When setting parameters γ_1 and γ_2, different purposes can be achieved based on the different effects of γ_1 and γ_2. γ_1 can be increased to allows larger step sizes because of decreasing the local cost, while increasing γ_2 enhances the importance of communicating the estimates of the global minimizer.

Remark 2. η_1 and η_2, although they are related to the global Laplace information, are not used in the design of the algorithm (23), and the conditions they fulfill are already established. Therefore, the designed algorithm still does not rely on the Laplace matrix information of the global network and is a fully distributed algorithm.

3.2. Distributed resourece allocation problem

Under Assumption 2 and 3, problem (20) has a unique solution x^*. There exists a unique $y^* \in \mathbb{R}^n$ which is the Lagrangian multiplier as follows,

$$\begin{aligned} \nabla g(x^*) + \mathbf{1}_N \otimes y^* = 0, \\ (\mathbf{1}_N^\top \otimes \mathbf{I}_n)(x^* - D) = 0, \end{aligned} \tag{44}$$

where $\nabla g(x) = \mathrm{col}(\nabla g_1(x_1), \ldots, \nabla g_N(x_N))$ and $D = \mathrm{col}(d_1, \ldots, d_N)$. In the same way, it is the KKT condition of the question (20). Specifically, given the Lagrangian function of problem (20), i. e., $L(x, y) = g(x) + y^\top(\mathbf{1}_N^\top \otimes \mathbf{I}_N)(x - D)$, the KKT condition (44) consists of $\nabla_x L(x, y) = 0$ and $\nabla_y L(x, y) = 0$.

The resource allocation problem (20) in a distributed system can be transformed into a consensus optimization problem using the KKT condition, which is regarded as a set of consensus constraints, i. e., the solution needs to satisfy all the KKT conditions so that the system reaches the consensus state. Consider the DST-based system resulting from incorporating a distributed integral feedback action of local dual variables as follows,

$${}^{\mathrm{C}}_{a}\nabla^{\alpha}_{k} x = -\kappa_1(\nabla g(x) + y), \tag{45a}$$

$$
\begin{aligned}
{}_{a}^{\mathrm{C}}\nabla_{k}^{\alpha} y &= x - D - (\mathcal{L}^{\beta} \otimes \mathbf{I}_n) y - (\mathcal{L} \otimes \mathbf{I}_n) z, && (45\text{b}) \\
{}_{a}^{\mathrm{C}}\nabla_{k}^{\alpha} z &= (\mathcal{L}^{\beta} \otimes \mathbf{I}_n) y, && (45\text{c})
\end{aligned}
$$

with dynamic coupling gains

$$
\beta_{ij} = \begin{cases} 0, & \text{if } e_{ji} \in \mathcal{E} \backslash \bar{\mathcal{E}}, \\ \bar{\beta}_{p+1,i_p}, & \text{if } e_{ji} \in \bar{\mathcal{E}}, \end{cases} \tag{46a}
$$

$$
{}_{a}^{\mathrm{C}}\nabla_{k}^{\alpha} \bar{\beta}_{p+1,i_p} = \kappa_2 [(y_{i_p} - y_{p+1}) - \textstyle\sum_{j \in \bar{\mathcal{N}}_{\mathrm{out}}(p+1)} (y_{p+1} - y_j)]^{\top} (y_{i_p} - y_{p+1}), \tag{46b}
$$

where $\kappa_1, \kappa_2 \in \mathbb{R}_+$ and $\mathcal{L}^{\beta}$ is the gain-dependent Laplacian matrix defined as

$$
\begin{aligned}
\mathcal{L}_{ij}^{\beta} &= -\beta_{ij} w_{ij}, \quad i \neq j, \\
\mathcal{L}_{ii}^{\beta} &= \sum\nolimits_{j=1, j \neq i}^{N} \beta_{ij} w_{ij}, \quad i = 1, \ldots, N.
\end{aligned}
$$

The product of the weight w_{ij} and the gain β_{ij} defines the feedback gain of the relative error vector $(y_i - y_j)$ for agent i when updating its states y_i and z_i. It's important to note that a_{ii} is not specified in (45) and (46a), as there are no self-loops in the system. According to (45), the gain β_{ij} is adjusted only when the edge $e_{ji} \in \bar{\mathcal{E}}$. This update process relies on agent i, agent j, and all the out-neighbors of agent i in DST, so it is distributed.

Theorem 3.3. Suppose Assumptions 2 and 3 hold. If $(\bar{x}, \bar{y}, \bar{z})$ is an equilibrium of (45) and x^* is the global minimizer of (20), then $(\bar{x}, \bar{y}) = (x^*, 1_N \otimes y^*)$.

Proof. Substituting $(\bar{x}, \bar{y}, \bar{z})$ into (45b), it follows that

$$
0 = \bar{x} - D - (\mathcal{L}^{\beta} \otimes \mathbf{I}_n) \bar{y} - (\mathcal{L} \otimes \mathbf{I}_n) \bar{z}. \tag{47}
$$

Since for any $\mathcal{L}^{\beta} \in \mathcal{M}_{\mathrm{r}}^{N}$, there exists $(\mathcal{L}^{\beta} \otimes \mathbf{I}_n) \bar{y} = 0$ such that $\bar{y} = \mathbf{1}_N \otimes y_0$, where $y_0 \in \mathbb{R}^n$ is some vector. Therefore, left-multiplying (47) by $(\mathbf{1}_N^{\top} \otimes \mathbf{I}_n)$, it can be obtained that

$$
\begin{aligned}
0 =& (\mathbf{1}_N^{\top} \otimes \mathbf{I}_n)(\bar{x} - D) - (\mathbf{1}_N^{\top} \otimes \mathbf{I}_n)(\mathcal{L}^{\beta} \otimes \mathbf{I}_n) \bar{y} - (\mathbf{1}_N^{\top} \otimes \mathbf{I}_n)(\mathcal{L} \otimes \mathbf{I}_n) \bar{z} \\
=& (\mathbf{1}_N^{\top} \otimes \mathbf{I}_n)(\bar{x} - D) - (\mathbf{1}_N^{\top} \mathcal{L} \otimes \mathbf{I}_n) \bar{z}.
\end{aligned} \tag{48}
$$

Given that $\mathbf{1}_N^{\top} \mathcal{L} = 0$, there has $(\mathbf{1}_N^{\top} \otimes \mathbf{I}_n)(\bar{x} - D) = 0$. Combining this with $\nabla g(\bar{x}) + \mathbf{1}_N \otimes y_0 = 0$, it can result in (44) and follows that $(\bar{x}, \bar{y}) = (x^*, \mathbf{1}_N \otimes y^*)$ exists and is unique.

Additionally, there are infinitely many solutions $\bar{z}$ that satisfy $(\mathcal{L} \otimes \mathbf{I}_m) \bar{z} = x - D$ because $\mathrm{rank}(\mathcal{L}) = N - 1$. In fact, if $(\bar{x}, \bar{y}, \bar{z})$ is an equilibrium of (45), then for any $\Delta z \in \mathbb{R}^n$, $(\bar{x}, \bar{y}, \bar{z} + \mathbf{1}_N \otimes \Delta z)$ is also an equilibrium of (45). □

Theorem 3.3 has completed the optimality analysis, indicating that the equilibrium point of (45) is the optimal solution to the problem (20). Next, Theorem 3.4 will explore the convergence of (45), proving whether the algorithm (45) can converge to the equilibrium point from any initial value.

Theorem 3.4. Under Assumptions 2 and 3, the adaptive algorithm (45) drives (x, y) to $(x^*, \mathbf{1}_N \otimes y^*)$ with Mittag–Leffler rate for any initial condition $x(a), y(a), z(a) \in \mathbb{R}^{Nn} \times \mathbb{R}^{Nn} \times \mathbb{R}^{Nn}$ and any $\beta_{ij}(a) \in \mathbb{R}$ with spanning-tree-based m-strongly convexity holding. Moreover, the adaptive gains $\bar{\beta}_{p+1,i_p}, p \in \mathcal{I}_{N-1}$, converge to some finite constant values.

The spanning-tree-based m-strongly convexity can be explained as follows. There exists a scalar $m \in \mathbb{R}_+$, such that the following condition (referred to as $\forall x, y \in \mathbb{R}^{Nn}$

$$(x-y)^\top(\bar{\mathcal{L}}^{\mathrm{U}} \otimes \mathbf{I}_n)(\nabla g(x) - \nabla g(y)) \geq m(x-y)^\top(\bar{\mathcal{L}}^{\mathrm{U}} \otimes \mathbf{I}_n)(x-y), \tag{49}$$

where $\bar{\mathcal{L}}^{\mathrm{U}} = \Xi^\top \Xi$ is the unweighted Laplacian matrix of the undirected spanning tree $\bar{\mathcal{G}}^{\mathrm{U}}$ based on $\bar{\mathcal{G}}$ (Ξ is defined as in Lemma 2.5).

P r o o f. The first step of the proof is to show that each trajectory of (45) converges to a equilibrium of (45).

Define the error vectors between the trajectory of (45) and any equilibrium $(\bar{x}, \bar{y}, \bar{z})$ of (45) as follows,

$$\mu = x - \bar{x}, \quad v = y - \bar{y}, \quad \eta = z - \bar{z}, \tag{50a}$$
$$\bar{\mu} = (\Xi \otimes \mathbf{I}_n)\mu, \quad \bar{v} = (\Xi \otimes \mathbf{I}_n)v, \quad \bar{\eta} = (\Xi \otimes \mathbf{I}_n)\eta. \tag{50b}$$

In a component-wise form, $\bar{\mu} = \mathrm{col}(\bar{\mu}_1, \ldots, \bar{\mu}_{N-1})$ where $\bar{\mu}_p = \mu_{i_p} - \mu_{p+1}$, $p \in \mathcal{I}_{N-1}$. Note that $\mathcal{L}^\beta \in \mathcal{M}_{\mathrm{r}}^N$.

According to (45), by using the statement 2) of Lemma 2.5, the properties of the Kronecker product and the fact that $(\Xi \otimes \mathbf{I}_n)\bar{y} = 0$, a new system comes out

$${}_a^{\mathrm{C}}\nabla_k^\alpha \bar{\mu} = -\kappa_1(\Xi \otimes \mathbf{I}_n)h - \kappa_1 \bar{v}, \tag{51a}$$
$${}_a^{\mathrm{C}}\nabla_k^\alpha \bar{v} = \bar{\mu} - (Q^\beta \otimes \mathbf{I}_n)\bar{v} - (Q \otimes \mathbf{I}_n)\bar{\eta}, \tag{51b}$$
$${}_a^{\mathrm{C}}\nabla_k^\alpha \bar{\eta} = (Q^\beta \otimes \mathbf{I}_n)\bar{v}, \tag{51c}$$
$${}_a^{\mathrm{C}}\nabla_k^\alpha \bar{\beta}_{p+1,i_p} = \kappa_2\big(\bar{v}_p - \sum\nolimits_{j \in \bar{\mathcal{N}}_{\mathrm{out}}(p+1)} \bar{v}_{j-1}\big)^\top \bar{v}_p, p \in \mathcal{I}_{N-1}, \tag{51d}$$

where $h = \nabla g(\mu + \bar{x}) - \nabla g(\bar{x})$, and Q (resp. Q^β), is defined as in Lemma 2.5 based on the DST $\bar{\mathcal{G}}$ and the (resp. gain-dependent) Laplacian matrix. More specifically, $Q^\beta = \tilde{Q}^\beta + \bar{Q}^\beta$ contains the fixed matrix $\tilde{Q}^\beta$ (note that ${}_a^{\mathrm{C}}\nabla_k^\alpha \beta_{ij} = 0$ if $e_{ji} \in \mathcal{E} \backslash \bar{\mathcal{E}}$), and the time-varying matrix

$$\bar{Q}_{pj}^\beta = \begin{cases} \bar{\beta}_{j+1,i_j} w_{j+1,i_j}, & \text{if } j = p, \\ -\bar{\beta}_{j+1,i_j} w_{j+1,i_j}, & \text{if } j = i_p - 1, \\ 0, & \text{otherwise.} \end{cases} \tag{52}$$

Consider the following candidate Lyapunov function

$$V_1 = \frac{1 + 3\bar{\lambda}(Q^\top Q)}{\epsilon_1 \underline{\lambda}^2(Q^{\mathrm{s}})} V_{\bar{\mu}} + V_{\bar{\nu}}^\beta + \frac{3\bar{\lambda}(Q^\top Q)}{\underline{\lambda}(Q^{\mathrm{s}})} V_{\bar{\eta}}, \tag{53}$$

where $V_{\bar{\mu}} = \frac{1}{2}\bar{\mu}^\top\bar{\mu}$, $V_{\bar{v}}^{\beta} = \frac{1}{2}\bar{v}^\top\bar{v}+\sum_{p=1}^{N-1}\frac{w_{p+1,i_p}}{2\kappa_2}(\bar{\beta}_{p+1,i_p}(t)-\phi_{p+1,i_p})^2$, $V_{\bar{\eta}} = \frac{1}{2}(\bar{v}+\bar{\eta})^\top(\bar{v}+\bar{\eta})$, and $Q^S > 0$ is guaranteed by 3) of Lemma 2.5, and $\epsilon_1, \phi_{p+1,i_p} \in \mathbb{R}_+, p = 1, \dots, N-1$, will be determined later.

According to Lemma 2.3, the fractional difference of $V_{\bar{\mu}}$ is shown as

$$
\begin{aligned}
{}_a^{\mathrm{C}}\nabla_k^{\alpha} V_{\bar{\mu}} &\le -\kappa_1\bar{\mu}^\top(\Xi\otimes\mathbf{I}_n)h - \kappa_1\bar{\mu}^\top\bar{\nu}, && \text{(54a)}\\
&\le -\kappa_1 m\bar{\mu}^\top\bar{\mu} - \kappa_1\bar{\mu}^\top\bar{v}, && \text{(54b)}\\
&\le -\kappa_1 m\bar{\mu}^\top\bar{\mu} + \epsilon_2\bar{\mu}^\top\bar{\mu} + \frac{\kappa_1^2}{4\epsilon_2}\bar{v}^\top\bar{v}, && \text{(54c)}\\
&\le (\epsilon_2 - \kappa_1 m)\bar{\mu}^\top\bar{\mu} + \frac{\kappa_1^2}{4\epsilon_2}\bar{v}^\top\bar{v}, && \text{(54d)}
\end{aligned}
$$

where $\epsilon_2 \in \mathbb{R}_+$ is to be decided later. The step in (54b) is derived from the fact that $\bar{\mu}^\top(\Xi\otimes\mathbf{I}_n)h \ge m\bar{\mu}^\top\bar{\mu}$ as known from (50a) and (49) and Young's inequality was used to get (54c).

According to Lemma 2.3, the fractional difference of $V_{\bar{v}}^{\beta}$ is formulated as

$$
\begin{aligned}
{}_a^{\mathrm{C}}\nabla_k^{\alpha} V_{\bar{v}}^{\beta} \le &\bar{v}^\top\bar{\mu} - \bar{v}^\top(Q^\beta\otimes\mathbf{I}_n)\bar{v} - \bar{v}^\top(Q\otimes\mathbf{I}_n)\bar{\eta}\\
&+\sum_{p=1}^{N-1} w_{p+1,i_p}(\bar{\beta}_{p+1,i_p} - \phi_{p+1,i_p})(\bar{v}_p - \sum_{j+1\in\bar{\mathcal{N}}_{\mathrm{out}}(p+1)}\bar{v}_j)^\top\bar{v}_p.
\end{aligned}
\tag{55}
$$

Define $\Phi \in \mathbb{R}^{(N-1)\times(N-1)}$ as

$$
\Phi_{pj} = \begin{cases} \phi_{j+1,i_j}w_{j+1,i_j}, & \text{if } j = p,\\ -\phi_{j+1,i_j}w_{j+1,i_j}, & \text{if } j = i_p - 1,\\ 0, & \text{otherwise.}\end{cases}
\tag{56}
$$

According to (52) and (56), it follows that

$$
w_{p+1,i_p}\bar{\beta}_{p+1,i_p} = \begin{cases} \bar{Q}_{pp}^{\beta}, & \text{if } j = p,\\ \bar{Q}_{jp}^{\beta}, & \text{if } j+1 \in \bar{\mathcal{N}}_{\mathrm{out}}(p+1),\end{cases}
\tag{57a}
$$

$$
w_{p+1,i_p}\phi_{p+1,i_p} = \begin{cases} \Phi_{pp}, & \text{if } j = p,\\ \Phi_{jp}, & \text{if } j+1 \in \bar{\mathcal{N}}_{\mathrm{out}}(p+1).\end{cases}
\tag{57b}
$$

From (57), it is inferred that

$$
\begin{aligned}
&\sum_{p=1}^{N-1} w_{p+1,i_p}(\bar{\beta}_{p+1,i_p} - \phi_{p+1,i_p})(\bar{v}_p - \sum_{j+1\in\bar{\mathcal{N}}_{\mathrm{out}}(p+1)}\bar{v}_j)^\top\bar{v}_p\\
=&\sum_{p=1}^{N-1}(\bar{Q}_{pp}^{\beta}\bar{v}_p + \sum_{j=1,j\ne p}^{N-1}\bar{Q}_{jp}^{\beta}\bar{v}_j)^\top\bar{v}_p\\
&-\sum_{p=1}^{N-1}(\Phi_{pp}\bar{v}_p + \sum_{j=1,j\ne p}^{N-1}\Phi_{jp}\bar{v}_j)^\top\bar{v}_p\\
=&\sum_{p=1}^{N-1}\sum_{j=1}^{N-1}(\bar{Q}_{jp}^{\beta} - \Phi_{jp})\bar{v}_j^\top\bar{v}_p\\
=&\,\bar{v}^\top[(\bar{Q}^{\beta} - \Phi)\otimes\mathbf{I}_n]\bar{v}.
\end{aligned}
\tag{58}
$$

Therefore, it implies from (58) that

$$\begin{aligned} {}_a^{\mathrm{C}}\nabla_k^{\alpha} V_{\bar{v}}^{\beta} \leq & \bar{v}^\top \bar{\mu} - \bar{v}^\top (Q^\beta \otimes \mathbf{I}_n)\bar{v} - \bar{v}^\top (Q \otimes \mathbf{I}_n)\bar{\eta} + \bar{v}^\top [(\bar{Q}^\beta - \Phi) \otimes \mathbf{I}_n]\bar{v}, \\ = & \bar{v}^\top \bar{\mu} - \bar{v}^\top ((\tilde{Q}^\beta + \Phi) \otimes \mathbf{I}_n)\bar{v} - \bar{v}^\top (Q \otimes \mathbf{I}_n)\bar{\eta}. \end{aligned} \tag{59}$$

The time-varying matrix $\bar{Q}^\beta$ has been eliminated in (59), and all the matrices remained are constant. Also using Young's inequality, it holds that

$$\begin{aligned} {}_a^{\mathrm{C}}\nabla_k^{\alpha} V_{\bar{v}}^{\beta} \leq & \bar{v}^\top \bar{\mu} - \bar{v}^\top ((\tilde{Q}^\beta + \Phi) \otimes \mathbf{I}_n)\bar{v} + \frac{\bar{v}^\top \bar{v}}{2} + \frac{\bar{\eta}^\top (Q^\top Q \otimes \mathbf{I}_n)\bar{\eta}}{2} \\ \leq & \frac{1}{\underline{\lambda}^2(Q^{\mathrm{s}})}\bar{\mu}^\top \bar{\mu} + (\frac{\underline{\lambda}^2(Q^{\mathrm{s}})}{4} + \frac{1}{2})\bar{v}^\top \bar{v} - \bar{v}^\top ((\tilde{Q}^\beta + \Phi) \otimes \mathbf{I}_n)\bar{v} \\ & + \frac{\bar{\lambda}(Q^\top Q)}{2}\bar{\eta}^\top \bar{\eta}. \end{aligned} \tag{60}$$

According to Lemma 2.3, the fractional difference of $V_{\bar{\eta}}$ is computed as

$${}_a^{\mathrm{C}}\nabla_k^{\alpha} V_{\bar{\eta}} \leq \bar{v}^\top \bar{\mu} - \bar{v}^\top (Q \otimes \mathbf{I}_n)\bar{\eta} + \bar{\eta}^\top \bar{\mu} - \bar{\eta}^\top (Q \otimes \mathbf{I}_n)\bar{\eta}. \tag{61}$$

Using Young's inequality, it follows that

$$\begin{aligned} {}_a^{\mathrm{C}}\nabla_k^{\alpha} V_{\bar{\eta}} \leq & \frac{1}{2\underline{\lambda}(Q^{\mathrm{s}})}\bar{\mu}^\top \bar{\mu} + \frac{\underline{\lambda}(Q^{\mathrm{s}})}{2}\bar{v}^\top \bar{v} + \frac{\bar{\lambda}(Q^\top Q)}{\underline{\lambda}(Q^{\mathrm{s}})}\bar{v}^\top \bar{v} + \frac{\lambda(Q^{\mathrm{s}})}{4}\bar{\eta}^\top \bar{\eta} \\ & + \frac{\lambda(Q^{\mathrm{s}})}{2}\bar{\eta}^\top \bar{\eta} + \frac{1}{2\underline{\lambda}(Q^{\mathrm{s}})}\bar{\mu}^\top \bar{\mu} - \underline{\lambda}(Q^{\mathrm{s}})\bar{\eta}^\top \bar{\eta} \\ \leq & \frac{1}{\underline{\lambda}(Q^{\mathrm{s}})}\bar{\mu}^\top \bar{\mu} + (\frac{\lambda(Q^{\mathrm{s}})}{2} + \frac{\bar{\lambda}(Q^\top Q)}{\underline{\lambda}(Q^{\mathrm{s}})})\bar{v}^\top \bar{v} - \frac{\lambda(Q^{\mathrm{s}})}{4}\bar{\eta}^\top \bar{\eta}. \end{aligned} \tag{62}$$

From (54), (60), (62), and (53), the fractional difference of V_1 along the trajectory of (51) is limited by

$$\begin{aligned} {}_a^{\mathrm{C}}\nabla_k^{\alpha} V_1 \leq & -\frac{(1 + 3\bar{\lambda}(Q^\top Q))(\kappa_1 m - \epsilon_1 - \epsilon_2)}{\epsilon_1 \underline{\lambda}^2(Q^{\mathrm{s}})}\bar{\mu}^\top \bar{\mu} \\ & - \bar{v}^\top [(\Phi^{\mathrm{s}} - \gamma \mathbf{I}_{N-1} + (\tilde{Q}^\beta)^{\mathrm{s}}) \otimes \mathbf{I}_n]\bar{v} - \frac{\bar{\lambda}(Q^\top Q)}{4}\bar{\eta}^\top \bar{\eta}, \end{aligned} \tag{63}$$

where $\gamma \in \mathbb{R}_+$is given by $\gamma = \frac{\kappa_1^2[1+3\bar{\lambda}(Q^\top Q)]}{4\epsilon_1\epsilon_2\underline{\lambda}^2(Q^{\mathrm{s}})} + \frac{3\bar{\lambda}^2(Q^\top Q)}{\underline{\lambda}^2(Q^{\mathrm{s}})} + \frac{3\bar{\lambda}(Q^\top Q)}{2} + \frac{\lambda^2(Q^{\mathrm{s}})}{4} + \frac{1}{2}$.

The aim is to make ${}_a^{\mathrm{C}}\nabla_k^{\alpha} V_1 \leq 0$ by choosing appropriate parameters ϵ_1, ϵ_2, and ϕ_{p+1,i_p}, $p = 1, \ldots, N-1$. Because Q is fixed, selecting ϵ_1 and ϵ_2 satisfying $\epsilon_1 + \epsilon_2 \leq \kappa_1 m$ such that $-\frac{(1+3\bar{\lambda}(Q^\top Q))(\kappa_1 m - \epsilon_1 - \epsilon_2)}{\epsilon_1 \underline{\lambda}^2(Q^{\mathrm{s}})}\bar{\mu}^\top \bar{\mu} \leq 0$. Because γ and $(\tilde{Q}^\beta)^{\mathrm{s}}$ are fixed, it only needs to choose appropriate ϕ_{p+1,i_p} such that the inequality $\Phi^{\mathrm{s}} - \bar{\gamma}\mathbf{I}_{N-1} > 0$ holds.

According to similar mathematical induction procedures in [36], for any positive real number $\bar{\gamma}$, there exists an appropriate choice of ϕ_{p+1,i_p}. Specifically, let

$$\phi_{2,\mathrm{i}_1} > \frac{\bar{\gamma}}{w_{2,\mathrm{i}_1}}, \phi_{p+1,i_p} > \bar{\gamma} + \frac{\sum_{j=2}^{p} \phi_{j,\mathrm{i}_{j-1}}^2 w_{j,\mathrm{i}_{j-1}}^2}{4w_{p+1,i_p}^{\lambda}(\Omega_{p-1})},$$

where $\Omega_1 = [\phi_{2,i_1} w_{2,i_1} - \bar{\gamma}]$, and $\Omega_p = \begin{bmatrix} \Omega_{p-1} & \varphi_p \\ \varphi_p^\top & \phi_{p+1,i_p} w_{p+1,i_p} - \bar{\gamma} \end{bmatrix}$ with $\varphi_p = \frac{1}{2}[\phi_{p1} w_{p1}, \phi_{p2} w_{p2}, \ldots, \phi_{p,p-1} w_{p,p-1}]^\top, p = 2, \ldots, N-1$. Then, the positive definiteness of $\Phi^{\mathrm{s}} - \bar{\gamma}\mathbf{I}_{N-1}(\Omega_{N-1})$ is guaranteed by the Schur complement and the induction principle.

Let $M' = [(\Phi^{\mathrm{s}} - \gamma \mathbf{I}_{N-1} + (\tilde{Q}^\beta)^{\mathrm{s}}) \otimes \mathbf{I}_n]$. From (63), it can be obtained that

$$\begin{aligned} {}_a^{\mathrm{C}}\nabla_k^\alpha V_1 \leq & -\frac{2(1+3\bar{\lambda}(Q^\top Q))(\kappa_1 m - \epsilon_1 - \epsilon_2)}{\epsilon_1 \underline{\lambda}^2(Q^{\mathrm{s}})}(\frac{1}{2}\bar{\mu}^\top \bar{\mu}) \\ & - 2\underline{\lambda}(M')(\frac{1}{2}\bar{v}^\top \bar{v}) - \frac{\bar{\lambda}(Q^\top Q)}{4}\bar{\eta}^\top \bar{\eta}. \end{aligned} \tag{64}$$

According to (64), the following inequalities all hold

$${}_a^{\mathrm{C}}\nabla_k^\alpha V_1 \leq -\frac{2(1+3\bar{\lambda}(Q^\top Q))(\kappa_1 m - \epsilon_1 - \epsilon_2)}{\epsilon_1 \underline{\lambda}^2(Q^{\mathrm{s}})}(\frac{1}{2}\bar{\mu}^\top \bar{\mu}), \tag{65a}$$

$${}_a^{\mathrm{C}}\nabla_k^\alpha V_1 \leq -2\underline{\lambda}(M')(\frac{1}{2}\bar{v}^\top \bar{v}), \tag{65b}$$

$${}_a^{\mathrm{C}}\nabla_k^\alpha V_1 \leq -\frac{\bar{\lambda}(Q^\top Q)}{2}(\frac{1}{2}\bar{\eta}^\top \bar{\eta}). \tag{65c}$$

Then, using Lemma 2.7, it can drive $(\bar{\mu}, \bar{v}, \bar{\eta}) \to (0,0,0)$ with Mittag–Leffler rate by selecting the appropriate ϵ_1, ϵ_2 and ϕ_{p+1,i_p}, and the adaptive gains $\bar{\beta}_{p+1,i_p}, p \in I_{N-1}$, converge to some finite constant values.

Returning to the original coordinates of (45), $(x,y,z) \to (\bar{x} + \mathbf{1}_N \otimes \Delta_x, \bar{y} + \mathbf{1}_N \otimes \Delta_y, \bar{z} + \mathbf{1}_N \otimes \Delta_z) := (x_{\mathrm{s}}, y_{\mathrm{s}}, z_{\mathrm{s}})$, where $\Delta_x, \Delta_y, \Delta_z \in \mathbb{R}^n$ are some deviation vectors.

Next step is to proof that $\Delta_x = \Delta_y = 0$ holds. The steady-state dynamics of Δ_x and Δ_y are formulated as follows,

$${}_a^{\mathrm{C}}\nabla_k^\alpha \Delta_x = \frac{1}{N}(\mathbf{1}_N^\top \otimes \mathbf{I}_n){}_a^{\mathrm{C}}\nabla_k^\alpha x_{\mathrm{s}}, \quad {}_a^{\mathrm{C}}\nabla_k^\alpha \Delta_y = \frac{1}{N}(\mathbf{1}_N^\top \otimes \mathbf{I}_n){}_a^{\mathrm{C}}\nabla_k^\alpha y_{\mathrm{s}}. \tag{66}$$

Substituting (45) evaluated at $(x_{\mathrm{s}}, y_{\mathrm{s}}, z_{\mathrm{s}})$ into the above equations, and noting that $O^\beta\big|_{(\bar{x},\bar{y},\bar{z})} = 0$, it can be obtained that

$${}_a^{\mathrm{C}}\nabla_k^\alpha \Delta_x = -\frac{\kappa_1}{N}(\mathbf{1}_N^\top \otimes \mathbf{I}_n)(\nabla g(x_{\mathrm{s}}) - \nabla g(\bar{x})) - \kappa_1 \Delta_y = 0, \tag{67a}$$

$${}_a^{\mathrm{C}}\nabla_k^\alpha \Delta_y = \Delta_x = 0. \tag{67b}$$

This implies that $\Delta_x = \Delta_y = 0$, i. e., $(x_{\mathrm{s}}, y_{\mathrm{s}}) = (\bar{x}, \bar{y})$. Thus, every trajectory of (45) converges to an equilibrium of (45). By Theorem 3.3, it follows that $(x,y) \to (x^*, \mathbf{1}_N \otimes y^*)$. □

In this case, the spanning-tree-based m-strongly convex condition (49) holds with any $m \leq \underline{\lambda}(\Theta)$ and for any DST. Immediately, it has the following corollary:

Corollary 3.5. Under Assumptions 2 and 3, the resource allocation problem (20) can be solved with the adaptive algorithm (45) for any initial conditions $(x(a), y(a), z(a) \in \mathbb{R}^{Nn} \times \mathbb{R}^{Nn} \times \mathbb{R}^{Nn}$ and any $\beta_{ij}(a) \in \mathbb{R}$, i. e., $(x,y) \to (x^*, \mathbf{1}_N \otimes y^*)$. Moreover, the adaptive gains $\bar{\beta}_{p+1,i_p}, p \in I_{N-1}$, converge to some finite constant values.

In this subsection, the fractional calculus and DST-based adapted frameworks are applied to solve the distributed resource allocation problem (20) and design a fractional distributed resource allocation algorithm with DST. It is verified its Mittag–Leffler convergence by Theorem 3.3 and Theorem 3.4 as the same as the last section.

The adaptive fractional distributed resource allocation algorithm proposed in this paper demonstrates significant advantages in solving distributed resource allocation problems:

i) The proposed fully distributed algorithm introduces a DST-based adaptive framework, eliminating the need for global Laplacian matrix information. Moreover, compared to the algorithm in [38], which requires strong convexity of local functions, the algorithm (45) only requires the global cost function to be convex, thereby relaxing the restrictions on local functions and enhancing the algorithm's applicability and flexibility.

ii) The proposed algorithm achieves Mittag–Leffler convergence, exhibiting faster convergence rates in fractional systems. Notably, when the fractional order $\alpha = 1$, the algorithm reduces to an integer algorithm while maintaining exponential convergence.

iii) The introduction of fractional calculus enriches the algorithm's dynamic characteristics, allowing it to better capture memory and hereditary properties, thereby improving convergence speed and optimization accuracy. By using the Nabla fractional calculus, the algorithm extends from a continuous-time framework to a discrete-time framework, and broadens its feasible parameter range.

4. NUMERICAL SIMULATIONS

4.1. Simulations of distributed consensus optimization algorithm with DST

The algorithm designed above is tested over a set of one-dimensional cost functions which are defined over $x \in \mathbb{R}$ as

$$
\begin{aligned}
f_1(x) &= 0.5\mathrm{e}^{-0.5x} + 0.4\mathrm{e}^{0.3x},\\
f_2(x) &= x^2 \ln(2 + x^2),\\
f_3(x) &= 0.5x^2 \ln(1 + x^2) + x^2,\\
f_4(x) &= x^2 + \mathrm{e}^{0.1x},\\
f_5(x) &= \ln(\mathrm{e}^{-0.1x} + \mathrm{e}^{0.3x}) + 0.1x^2,\\
f_6(x) &= (1 + \mathrm{e}^{x})^{-1}.
\end{aligned}
$$

Consider the following multi-agent network topology in Figure 1.

To verify the Theorem (3.2), choose DST $\bar{\mathcal{G}}$ which is red highlighted in Figure 1, and the parameters in the algorithm are chosen as $\gamma_1 = \gamma_2 = 0.5$. Choose initial values $x(a)$, $y(a)$ selected from the standard Gaussian distribution. The fractional order is chosen as $\alpha = 0.8$.

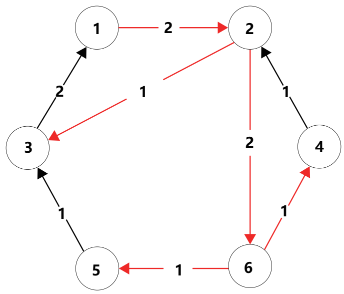
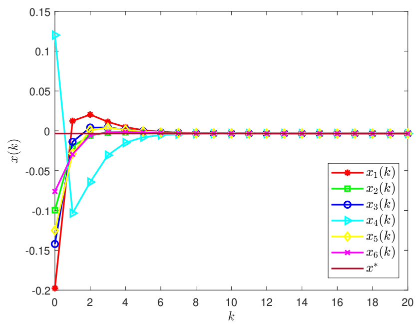
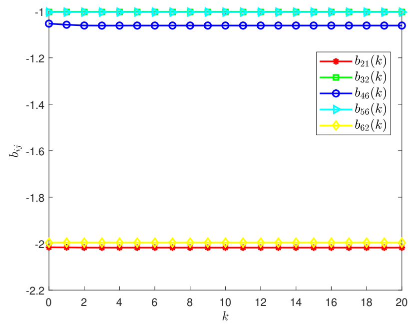


Fig. 1. Balanced digraph $\mathcal{G}$ with DST $\bar{\mathcal{G}}$, highlighted in red.

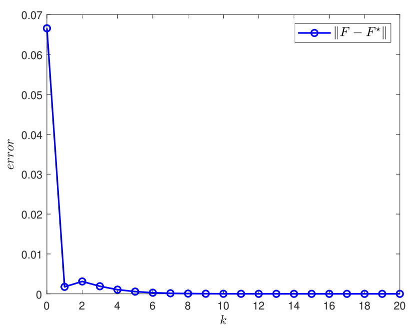


(a) The local estimates $x_i(k)$

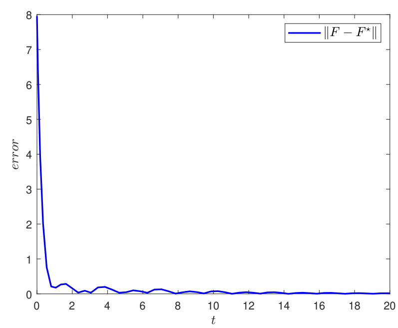


(b) The parameters $\bar{b}_{p+1,i_p}$

Fig. 2. The trajectories of the local estimates $x_i(k)$ in (a) and the parameters $\bar{b}_{p+1,i_p}$ in (b).



(a) Fractional discrete algorithm



(b) Integer continuous algorithm

Fig. 3. Comparison of optimization value errors generated by integer algorithm in [7] and fractional algorithm (23).

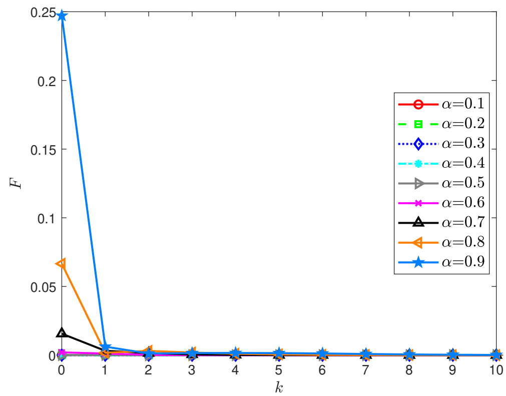


Fig. 4. The trajectories of the global cost function F with different fractional order α.

In the Figure 2(a), it can see x converges to the global minimum point x^*, while the parameters $\bar{b}_{p+1,i_p}$, $p \in \mathcal{I}_{N-1}$ converge to fixed constant values in the Figure 2(b).

Figure 3 is provided to compare the integer continuous algorithm [7] with the fractional discrete algorithm (23) presented in this paper, both using the digraph in the Figure 1 and cost functions above. The optimal value error curves for the integer and fractional algorithms are given respectively. This shows that the algorithm in this paper can achieve the same convergence performance as the integer algorithm, and it also has a performance advantage in the comparison that the errors' trajectory of the fractional algorithm is smoother, and convergence faster stabilizes at zero. Also, in Figure 4 it provided a numerical simulation with different fractional order $\alpha \in (0, 1]$. It can be observed that convergence is still achieved.

4.2. Simulations of distributed resource allocation algorithm with DST

Consider $N = 6$ agents communicating over a new balanced digraph Figure 5. There is a total resource d and each agent has its local resource d_i equally divided by d, which $d_i = d/N$. The local cost function $g_i(x) = 0.1x^2 + t_i x$ is associated to each agent i, and t_i, for all i is a random number selected in $[1, 100]$. Let the initial $(x(a)), y(a), z(a))$ randomly chosen from a Gaussian distribution, and the initial $\beta(a)$ is chosen randomly between $[0, 1]$.

Consider the fractional order $\alpha = 0.8$. Chose $\kappa_1 = \kappa_2 = 1$ and $d = 1.5 \times 10^3$. The local estimates x_i of agents and the dynamic coupling gains β under (45) and (46a) are provided in Figure 6. Only $\bar{\beta}_{p+1,i_p}$ updates in β. To compare with the first case, the states of agents and coupling gains on DST are provided in Figure 7 with $\kappa_1 = 10$, $\kappa_2 = 0.1$. It shows that a lager κ_1 leads to better transient performance of x_i and a small κ_2 leads to smaller steady values of $\bar{\beta}_{p+1,i_p}$.

Compare the $\alpha = 0.8$ fractional discrete algorithm and the integer continuous algorithm [38] with $\kappa_1 = \kappa_2 = 1$ and the result is provided in Figure 8 and Figure 9. By comparison, it is clear that the convergence curve of the fractional algorithm is more sta-

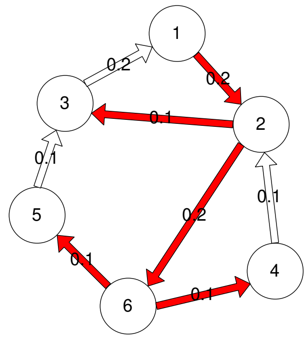
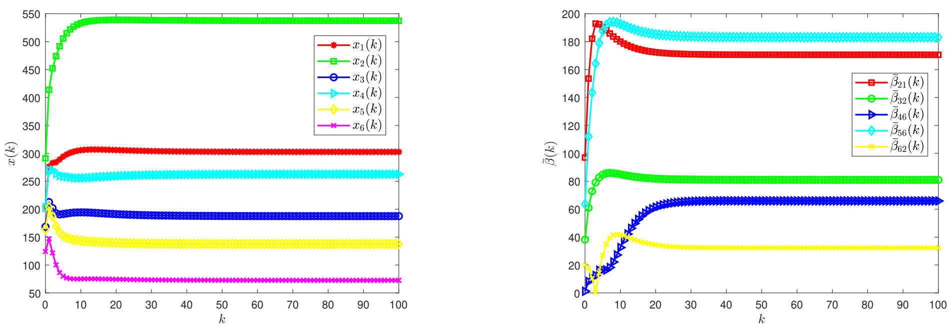


Fig. 5. Balanced digraph $\mathcal{G}$ with DST $\bar{\mathcal{G}}$, highlighted in red.



(a) The local estimates $x_i(k)$

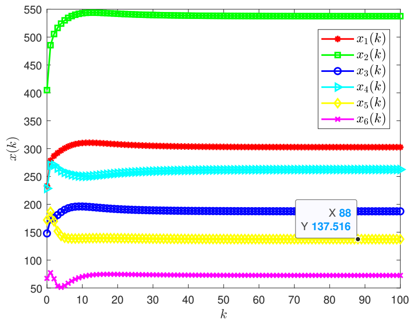
(b) The coupling gains $\bar{\beta}_{p+1,i_p}$

Fig. 6. The local estimates $x_i(k)$ and the coupling gains $\bar{\beta}_{p+1,i_p}$ with $\kappa_1 = \kappa_2 = 1$.

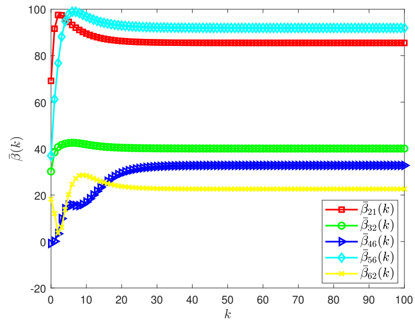
ble than that of the integer algorithm, so the fractional algorithm has better robustness and anti-interference ability. The fractional algorithm also improves the convergence rate of $\bar{\beta}_{p+1,i_p}$. In particular, within a certain simulation period $k = t = 100$, the integer algorithm requires 1383 iteration points, whereas the fractional algorithm needs only 100 iteration points. This comparison highlights the discrete nature of the nabla fractional algorithm, which demands fewer iteration points or updates while maintaining the same level of accuracy.

To furnish additional examples, the article presents simulations under the fractional order $\alpha = 0.9$ in Figure 10, demonstrating that convergence can likewise be achieved under these conditions.

To furnish additional examples, the article presents simulations under the fractional order $\alpha = 0.9$ in Figure 10, demonstrating that convergence can likewise be achieved under these conditions.

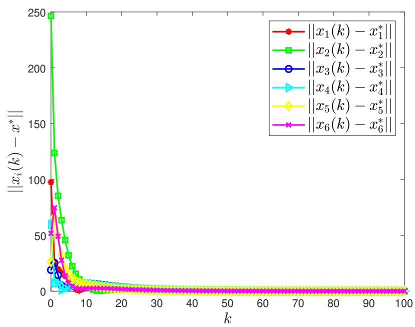


(a) The local estimates $x_i(k)$

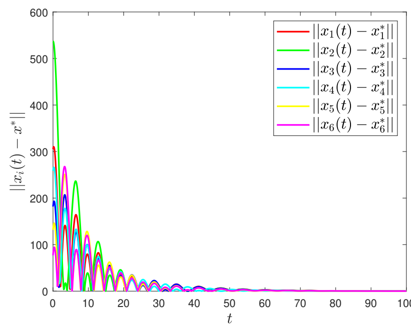


(b) The coupling gains $\bar{\beta}_{p+1,i_p}$

Fig. 7. The local estimates $x_i(k)$ and the coupling gains $\bar{\beta}_{p+1,i_p}$ with $\kappa_1 = 10, \kappa_2 = 0.1$.



(a) Fractional discrete algorithm

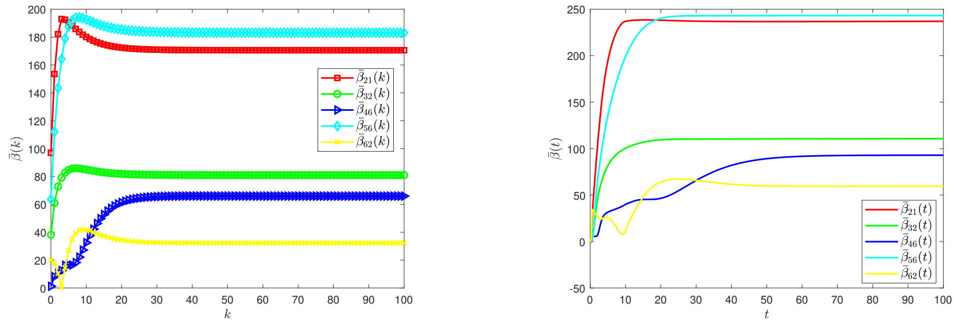


(b) Integer continuous algorithm

Fig. 8. Comparison of the state $||x_i(k) - x^*||$ of agents generated by fractional algorithm (45) and integer algorithm in [38].

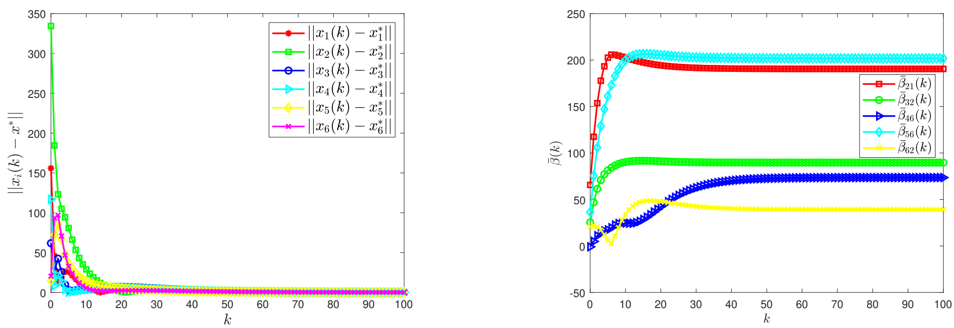
5. CONCLUSIONS

In this paper, two fully distributed fractional algorithms based on DST are proposed to address optimization and resource allocation problems. The Mittag–Leffler convergence of both algorithms is rigorously analyzed, and their performance is validated through extensive simulations. Fitst, the proposed DST-based fully distributed algorithms eliminate the need for global Laplacian matrix information, and avoids the requirement for sufficiently small step sizes. Secondly, by incorporating fractional calculus, the algorithms achieve improved performance and reduced communication costs. Moreover, both algorithms relax the convexity condition, requiring only the global cost function to be convex while allowing local cost functions to be non-convex. Future research will focus on extending the framework to unbalanced digraphs, further reducing dependencies on directed spanning tree matrix information and exploring the effects of



(a) Fractional discrete algorithm (b) Integer continuous algorithm

Fig. 9. Comparison of the gains $\bar{\beta}_{p+1,i_p}$ of agents generated by fractional algorithm (45) and integer algorithm in [38].



(a) The state $\|x_i(k) - x^*\|$ (b) The coupling gains $\bar{\beta}_{p+1,i_p}$

Fig. 10. The trajectories of the state $\|x_i(k) - x^*\|$ in (a) and the gains $\bar{\beta}_{p+1,i_p}$ in (b) with $\alpha = 0.9$.

introducing noise to create controlled interference in the algorithms.

ACKNOWLEDGEMENT

This work is supported by the National Natural Science Foundation of China (62273092), the Guangxi Natural Science Foundation (2025GXNSFDA069040), the Jiangsu Provincial Scientific Research Center of Applied Mathematics (BK20233002), the Fundamental Research Funds for the Central Universities (4207012301, 2025SMECP07), and the Guangdong Provincial Key Laboratory of Mathematical Foundations for Artificial Intelligence (2023B1212010001).

(Received September 19, 2024)

REFERENCES

[1] S. Boyd, N. Parikh, E. Chu, B. Peleato, J. Eckstein et al.: Distributed optimization and statistical learning via the alternating direction method of multipliers. Found. Trends Machine Learn. *3* (2011), 1, 1–122.

[2] F. Bullo, J. Cortés, and S. Martinez: Distributed control of robotic networks: a mathematical approach to motion coordination algorithms. Princeton University Press, Princeton 2009.

[3] Y. Q. Chen, Q. Gao, Y. H. Wei, and Y. Wang: Study on fractional order gradient methods. Appl. Math. Comput. *314* (2017), 310–321. DOI:10.1016/j.amc.2017.07.023

[4] S. S. Cheng and S. Liang: Distributed optimization for multi-agent system over unbalanced graphs with linear convergence rate. Kybernetika *56* (2020), 3, 559–577. DOI:10.14736/kyb-2020-3-0559

[5] S. S. Cheng, S. Liang, and Y. Fan: Distributed solving Sylvester equations with fractional order dynamics. Control Theory Technol. *19* (2021), 2, 249–259. DOI:10.1007/s11768-021-00044-0

[6] D. R. Ding, Q. L. Han, and X. H. Ge: Distributed filtering of networked dynamic systems with non-gaussian noises over sensor networks: a survey. Kybernetika *56* (2020), 1, 5–34.

[7] B. Gharesifard and J. Cortés: Distributed continuous-time convex optimization on weight-balanced digraphs. IEEE Trans. Automat. Control *59* (2014), 3, 781–786. DOI:10.1109/TAC.2013.2278132

[8] X. L. Hong, Y. H. Wei, S. Y. Zhou, and D. D. Yue: Nabla fractional distributed optimization algorithms over undirected/directed graphs. J. Franklin Inst. *361* (2024), 3, 1436–1454. DOI:10.1016/j.jfranklin.2024.01.013

[9] J. Y. Huang, S. Y. Zhou, H. Tu, Y. H. Yao, and Q. S. Liu: Distributed optimization algorithm for multi-robot formation with virtual reference center. IEEE/CAA J. Automat. Sinica *9* (2022), 4, 732–734. DOI:10.1109/JAS.2022.105473

[10] P. Humblet: A distributed algorithm for minimum weight directed spanning trees. IEEE Trans. Commun. *31* (1983), 6, 756–762. DOI:10.1109/TCOM.1983.1095883

[11] S. S. Kia, J. Cortés, and S. Martínez: Distributed convex optimization via continuous-time coordination algorithms with discrete-time communication. Automatica *55* (2015), 254–264. DOI:10.1016/j.automatica.2015.03.001

[12] Z. H. Li, Z. T. Ding, J. Y. Sun, and Z. K. Li: Distributed adaptive convex optimization on directed graphs via continuous-time algorithms. IEEE Trans. Automat. Control *63* (2018), 5, 1434–1441. DOI:10.1109/TAC.2017.2750103

[13] S. Liang, L. Y. Wang, and G. Yin: Fractional differential equation approach for convex optimization with convergence rate analysis. Optimiz. Lett. *45* (2019), 9, 145–155.

[14] P. Lin, W. Ren, and J. A. Farrell: Distributed continuous-time optimization: Nonuniform gradient gains, finite-time convergence, and convex constraint set. IEEE Trans. Automat. Control *62* (2017), 5, 2239–2253. DOI:10.1109/TAC.2016.2604324

[15] Q. S. Liu, S. F. Yang, and Y. G. Hong: Constrained consensus algorithms with fixed step size for distributed convex optimization over multiagent networks. IEEE Trans. Automat. Control *62* (2017), 8, 4259–4265. DOI:/10.1109/TAC.2017.2681200

[16] D. K. Molzahn, F. Dörfler, H. Sandberg, S. H. Low, S. Chakrabarti, R. Baldick, and J. Lavaei: A survey of distributed optimization and control algorithms for electric power systems. IEEE Trans. Smart Grid *8* (2017), 6, 2941–2962. DOI:10.1109/TSG.2017.2720471

[17] A. Nedić and A. Ozdaglar: Distributed subgradient methods for multi-agent optimization. IEEE Trans. Automat. Control *54* (2009), 1, 48–61. DOI:10.1109/tac.2008.2009515

[18] W. Ni and X. L. Wang: Averaging approach to distributed convex optimization for continuous-time multi-agent systems. Kybernetika *52* 2016), 6, 898–913. DOI:10.14736/kyb-2016-6-0898

[19] X. T. Ni, Y. H. Wei, S. Y. Zhou, and M. Tao: Multi-objective network resource allocation method based on fractional PID control. Signal Process. *227* (2025), 109717. DOI:10.1016/j.sigpro.2024.109717

[20] S. Pu, W. Shi, J. Xu, and A. Nedic: Pushpull gradient methods for distributed optimization in networks. IEEE Trans. Automat. Control *66* (2021), 1, 1–16.

[21] Y. F. Pu, J. L. Zhou, Y. Zhang, N. Zhang, G. Huang, and P. Siarry: Fractional extreme value adaptive training method: fractional steepest descent approach. IEEE Trans. Neural Networks Learn. Systems *26* (2015), 4, 653–662. DOI:10.1109/TNNLS.2013.2286175

[22] W. Ren and Y. C. Cao: Distributed Coordination of Multi-Agent Networks: Emergent Problems, Models, and Issues. Springer Science and Business Media, 2010.

[23] Y. W. Song, J. D. Cao, and L. Rutkowski: A fixed-time distributed optimization algorithm based on event-triggered strateg. IEEE Trans. Network Sci. Engrg. *9* (2021), 3, 1154–1162. DOI:10.1109/TNSE.2021.3133541

[24] B. Touri and B. Gharesifard: A modified saddle-point dynamics for distributed convex optimization on general directed graphs. IEEE Trans. Automat. Control *65* (2020), 7, 3098–3103. DOI:10.1109/TAC.2019.2947184

[25] D. Varagnolo, F. Zanella, A. Cenedese, G. Pillonetto, and L. Schenato: Newton-Raphson consensus for distributed convex optimization. IEEE Trans. Automat. Control *61* (2016), 4, 994–1009. DOI:10.1109/tac.2015.2449811

[26] D. Wang and Z. Z. Gao: Distributed finite-time optimization algorithms with a modified Newton–Raphson method. Neurocomputing *536* (2023), 73–79. DOI:10.1016/j.neucom.2023.03.027

[27] Y. H. Wang, P. Lin, and H. S. Qin: Distributed classification learning based on nonlinear vector support machines for switching networks. Kybernetika *53* (2017), 4, 595–611. DOI:10.14736/kyb-2017-4-0595

[28] X. Y. Wang, G. D. Wang, and S. H. Li: Distributed finite-time optimization for integrator chain multiagent systems with disturbances. IEEE Trans. Automat. Control *65* (2020), 12, 5296–5311. DOI:10.1109/TAC.2020.2979274

[29] Y. H. Wei and Y. Q. Chen: Converse Lyapunov theorem for nabla asymptotic stability without conservativeness. IEEE Trans. Systems Man Cybernet.: Systems *52* (2022), 4, 2676–2687. DOI:10.1109/TSMC.2021.3051639

[30] Y. H. Wei, Y. Kang, W. D. Yin, and Y. Wang: Generalization of the gradient method with fractional order gradient direction. J. Franklin Inst. *357* (2020), 4, 2514–2532. DOI:10.1016/j.jfranklin.2020.01.008

[31] Y. D. Wei, Y. H. Wei, Y. Q. Chen, and Y. Wang: Mittag–Leffler stability of nabla discrete fractional order dynamic systems. Nonlinear Dynamics *101* (2020), 407–417. DOI:10.1007/s11071-020-05776-3

[32] Y. H. Wei, L. L. Zhao, X. Zhao, and J. D. Cao: Fractional difference inequalities for possible Lyapunov functions: A review. Fract. Calculus Appl. Anal. (2024). DOI:10.1007/s13540-024-00298-w

[33] R. Xin and U. A. Khan: A linear algorithm for optimization over directed graphs with geometric convergence. IEEE Control Systems Lett. *2* (2018), 3, 315–320. DOI:10.1109/LCSYS.2018.2834316

[34] X. L. Yang, W. M. Zhao, J. X. Yuan, T. Chen, C. Zhang, and L. Q. Wang: Distributed optimization for fractional-order multi-agent systems based on adaptive backstepping dynamic surface control technology. Fractal Fractional *6* (2022), 11, 642. DOI:10.3390/fractalfract6110642

[35] D. D. Yue, S. Baldi, J. D. Cao, and B. De Schutter: Distributed adaptive optimization with weight-balancing. IEEE Trans. Automat. Control *67* (2022), 4, 2068–2075. DOI:10.1109/TAC.2021.3071651

[36] D. D. Yue, S. Baldi, J. D. Cao, Q. Li, and B. De Schutter: A directed spanning tree adaptive control solution to time-varying formations. IEEE Trans. Control Network Syst. *8* (2021), 2, 690–701. DOI:10.1109/TCNS.2021.3050332

[37] Y. K. Zeng, Y. H. Wei, S. Y. Zhou, and D. D. Yue: Distributed optimization via active disturbance rejection control: a nabla fractional design. Kybernetika *60* (2024), 1, 90–109. DOI:10.14736/kyb-2024-1-0090

[38] D. D. Yue, S. Baldi, J. D. Cao, Q. Li, and B. De Schutter: Distributed adaptive resource allocation: an uncertain saddle-point dynamics viewpoint. IEEE/CAA J. Automat. Sinica *10* (2023), 12, 2209–2221. DOI:10.1109/JAS.2023.123402

[39] J. Zhang, L. Liu, X. H. Wang, and H. B. Ji: Fully distributed algorithm for resource allocation over unbalanced directed networks without global Lipschitz condition. IEEE Trans. Automat. Control *68* (2022), 8, 5119–5126. DOI:10.1109/TAC.2022.3216972

[40] Y. L. Zheng and Q. S. Liu: A review of distributed optimization: Problems, models and algorithms. Neurocomputing *483* (2022), 446–459. DOI:10.1016/j.neucom.2021.06.097

[41] S. Y. Zhou, Y. H. Wei, S. Liang, and J. Cao: A gradient tracking protocol for optimization over nabla fractional multi-agent systems. IEEE Trans. Signal Inform. Process. Networks *10* (2024), 500–512. DOI:10.1109/TSIPN.2024.3402354

Huaijin Peng, School of Mathematics, Southeast University, Nanjing 211189. P. R. China.
e-mail: 213212687@seu.edu.cn

Yiheng Wei, Corresponding author. School of Mathematics, Southeast University, Nanjing 211189. P. R. China.
e-mail: neudawei@seu.edu.cn

Shuaiyu Zhou, School of Mathematics, Southeast University, Nanjing 211189. P. R. China.
e-mail: sy_zhou@seu.edu.cn

Dongdong Yue, College of Electrical Engineering and Control Science, Nanjing Tech University, Nanjing 211816. P. R. China.
e-mail: yued@njtech.edu.cn