



# 一类基于分数阶梯度信息的变阶次扩散 LMS 算法

杨洋<sup>1</sup>, 莫立坡<sup>1,2\*</sup>, 左敏<sup>2</sup>, 于永光<sup>3</sup>

1. 北京工商大学数学与统计学院, 北京 100048

2. 北京工商大学计算机与人工智能学院, 北京 100048

3. 北京交通大学数学与统计学院, 北京 100044

\* 通信作者. E-mail: molipo@th.btbu.edu.cn

收稿日期: 2024-01-03; 修回日期: 2024-03-24; 接受日期: 2024-05-29; 网络出版日期: 2024-08-08

北京市属高等学校高水平科研创新团队建设支持计划项目 (批准号: BPHR20220104) 和国家自然科学基金 (批准号: 61973329) 资助项目

**摘要** 针对分布式网络中的参数估计问题, 本文提出了一类基于分数阶梯度信息的扩散 LMS 算法, 主要利用分数阶梯度的变阶次机制来提升算法的各项性能. 首先, 针对已有的集中式分数阶梯度 LMS 算法, 将其推广到分布式网络中的参数估计问题上来. 进而, 讨论了所提算法的收敛速度和收敛精度. 其次, 考虑到分数阶阶次对于算法性能的影响, 设计了一个变分数阶阶次的策略来充分发挥分数阶的优点以提升算法的收敛特性. 进一步, 证明了切换拓扑结构下所提算法的收敛性. 最后, 通过数值仿真结果, 从收敛速度、收敛精度、鲁棒性等角度验证了所提算法的有效性和优越性.

**关键词** 分布式估计, 扩散式 LMS 算法, 适应性滤波, 切换拓扑, 分数阶微积分

## 1 引言

近些年来, 关于分布式网络的估计和控制问题在学界引起了持续的兴趣, 尤其是在分布式参数估计领域<sup>[1~4]</sup>. 相比集中式算法, 分布式算法不需要中心服务器, 具有更强的鲁棒性, 还可以节省通信成本<sup>[5]</sup>. 因此, 分布式参数估计已经被广泛应用于模式识别、噪声消除以及生物模型等诸多实际场景中<sup>[6~8]</sup>.

通常一个分布式网络中包含大量的节点, 其中每个节点均具有一定的调节自身参数的适应能力, 然而单个节点可能不会快速准确地估计出待估参数. 所谓分布式参数估计是指网络中的各个节点通过信息交互共享合作完成参数估计任务<sup>[9]</sup>. 根据节点在网络中传输信息的内容和方式的不同, 学者们提出了不同类型的分布式算法, 例如一致性 LMS 策略、增量式 LMS 策略以及扩散式 LMS 策略等<sup>[10~13]</sup>.

**引用格式:** 杨洋, 莫立坡, 左敏, 等. 一类基于分数阶梯度信息的变阶次扩散 LMS 算法. 中国科学: 信息科学, 2024, 54: 1907–1923, doi: 10.1360/SSI-2024-0003  
Yang Y, Mo L P, Zuo M, et al. A class of diffusion LMS algorithm with variable fractional order gradient (in Chinese). Sci Sin Inform, 2024, 54: 1907–1923, doi: 10.1360/SSI-2024-0003

相比一致性策略和增量式策略, 扩散式策略具有更大的稳定性范围和更强的整体鲁棒性<sup>[14]</sup>. 事实上, 扩散式策略对于一些复杂环境中数据的变化反映更加迅速, 对于节点和通信的失败具有更强的包容性. 在扩散参数估计策略中, 节点首先与其邻居节点交换估计值, 再通过组合邻居信息和自身信息更新对参数的估计. 根据适应和组合的顺序不同, 有两种不同类型的扩散式策略, 即先组合后适应 (combination then adaptive, CTA) 或先适应后组合 (adaptive then combination, ATC) 策略<sup>[15]</sup>. 对于扩散式 LMS 算法 (diffusion least mean square, DLMS) 的改进, 目前主要有三方面的研究, 其一是算法的迭代方式的改进, 例如通过添加正则项来改善算法收敛精度和速度的正则扩散 LMS 算法 (normal diffusion least mean square, NDLMS)<sup>[16~18]</sup>; 利用正割函数来避免算法陷入局部最优值的 Q 梯度的扩散 LMS 算法 (Q diffusion least mean square, QDLMS)<sup>[19]</sup>; 利用数据块作为运算单位以提升整体算法运算精度的仿射投影扩散 LMS 算法<sup>[20]</sup>. 其二是网络各节点信息通信权重系数的改进, 一般来讲, 依赖常数权重系数的网络在收敛速度方面有优势, 而自适应的网络权重在算法的收敛精度方面表现更加优秀. 例如自适应方差准则 (adaptive read variance, ARV) 通过利用节点噪声方差信息更新网络通信权重来改善算法的均方性能<sup>[21]</sup>. 其三是关于算法步长方面的研究, 主要包含利用自适应机制来减少算法对于初始步长依赖的变步长算法 (varying step search diffusion least mean square, VSSDLMS), 以及基于最优步长选择的可变步长算法<sup>[22~24]</sup>.

此外, 由于全局性和记忆性的特点, 分数阶梯度已经被引入集中式的 LMS 算法以提高算法的性能<sup>[25~27]</sup>. 分数阶算法的另一个优势在于阶次的可调节性. 一般来说, 较大的分数阶阶次可以显著加快算法的收敛速度, 而较小的分数阶阶次可以提高算法的收敛精度, 也就是减少 LMS 算法中的均方误差. 分布式网络中的每个节点可以根据个体误差, 调节自身的分数阶阶次, 以彰显分布式系统定制的特点<sup>[28]</sup>. 分数阶梯度为算法设计提供了更大的设计自由度, 开发基于分数阶梯度信息的扩散式 LMS 算法是一个具有前景的研究方向. 然而, 前面的研究仅仅局限于固定拓扑上的集中式分数阶 LMS 算法上, 采用了固定步长, 而在实际场景中, 固定步长不能时时满足分数阶算法的收敛条件. 因此, 本文通过设计可变分数阶阶次的策略来弥补固定步长造成的缺陷, 同时基于纯确定性框架, 通过简化线性时变系统 (linear time varying, LTV) 的稳定性分析<sup>[29]</sup>, 来研究切换拓扑下算法的收敛性.

本文主要利用分数阶梯度信息对经典 LMS 算法进行改进, 从而将已有的集中式分数阶 LMS 算法推广到分布式网络之上. 同时设计了一种基于分数阶梯度信息的变阶次扩散 LMS 算法并研究了其收敛性能. 本文的主要贡献可以总结如下:

- (1) 将集中式的分数阶 LMS 算法扩展到分布式的情形. 对于 ATC 策略下的扩散式 LMS 算法, 分析了整个分布式网络的收敛条件、收敛速度、收敛精度以及分数阶阶次的影响.
- (2) 根据分数阶阶次与算法性能的关系, 设计了两类变分数阶策略, 以最大程度利用分数阶梯度信息的优势.
- (3) 不同于已有的基于固定通信拓扑的结果, 本文还讨论了切换拓扑下算法的收敛性问题. 在网络拓扑包含一颗联合生成树的条件下, 证明了算法的收敛性.

**符号说明.** 在本文中,  $\mathcal{N}$  为自然数集,  $\Gamma(\cdot)$  代表伽马函数,  $\otimes$  代表矩阵间的克罗内克 (Kronecker) 积,  $\|\mathbf{A}\|$  表示向量或矩阵的欧几里得 (Euclid) 范数,  $\mathbf{A}^T$  为向量或矩阵的转置,  $\text{vec}(\mathbf{A})$  表示矩阵  $\mathbf{A}$  的向量化.

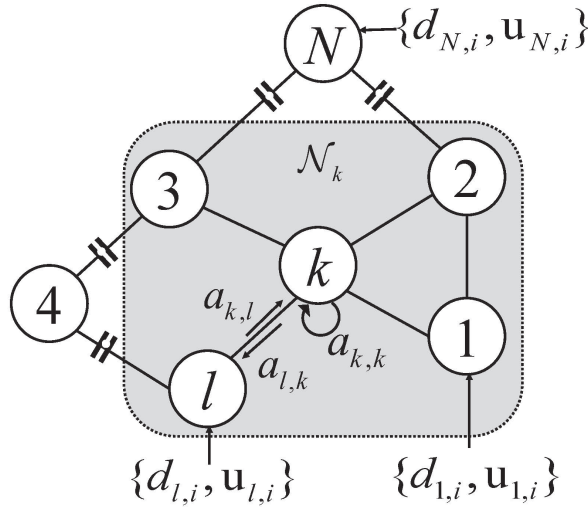


图 1 节点间信息的传输

Figure 1 Information exchange between nodes

## 2 问题描述和算法设计

### 2.1 问题描述

本文考虑一个由  $N$  个节点构成的分布式网络, 协同估计  $M \times 1$  维参数  $\mathbf{w}^0$ . 假设在每个时刻  $i$  各节点均可以独立获得自身带噪声的测量值  $d_k(i)$ , 以及一个维数为  $1 \times M$  的输入回归向量  $\mathbf{u}_{k,i}$ , 节点间信息的传输如图 1 所示, 具体的关系式可以表述为

$$d_k(i) = \mathbf{u}_{k,i} \mathbf{w}^0 + v_k(i), \quad k = 1, \dots, N, \quad (1)$$

其中  $v_k(i)$  是一个均值为 0 方差为  $\sigma_{v,k}^2$  的时空独立的高斯 (Gauss) 噪声. 在持续激励假设下, 通常将估计全局最优参数  $\mathbf{w}_0$  的目标函数转变为优化局部最优损失函数<sup>[1]</sup>, 即

$$J_k^g(\mathbf{w}_{k,i}) = \sum_{l \in \mathcal{N}_k} c_{lk} J_l^{\text{loc}}(\mathbf{w}_{l,i}) + \sum_{l \in \mathcal{N}_k/k} b_{lk} \|\mathbf{w}_{l,i} - \mathbf{w}^0\|^2, \quad (2)$$

其中  $J_k^{\text{loc}}(\mathbf{w}_{k,i}) = (d_k(i) - \mathbf{u}_{k,i} \mathbf{w}_{k,i})^2$  是节点的局部目标函数, 而  $b_{lk}$  是非负权重系数,  $\mathcal{N}_k$  是节点  $k$  的邻居节点的集合, 而  $c_{lk}$  是满足下列条件的非负权重:

$$\sum_{l \in \mathcal{N}_k} c_{lk} = 1, \quad c_{lk} = 0, \quad l \notin \mathcal{N}_k. \quad (3)$$

本文的目标是通过设计扩散式分数阶 LMS 算法对参数  $\mathbf{w}^0$  作出估计.

### 2.2 算法设计

分数阶导数有着多种定义, 其中 Caputo 定义由于其具有整数阶初值的特性, 使用较多, Caputo 分数阶导数定义如下:

$${}_{t_0}^C \mathcal{D}_t^\alpha f(t) = \frac{1}{\Gamma(n - \alpha)} \int_{t_0}^t \frac{f^{(n)}(\tau)}{(t - \tau)^{\alpha - n + 1}} d\tau, \quad (4)$$

其中  $n-1 < \alpha < n, n \in \mathcal{N}$ ,  $f^{(n)}(\tau)$  为函数的  $n$  阶整数阶导数. 可以证明, 式 (4) 能够展成泰勒 (Taylor) 级数的形式 [30]:

$${}^C_{t_0} \mathcal{D}_t^\alpha f(t) = \sum_{i=1}^{\infty} \frac{f^{(i)}(t_0)}{\Gamma(i+1-\alpha)} (t-t_0)^{i-\alpha}, \quad 0 < \alpha < 1. \quad (5)$$

**注释1** Riemman-Liouville 型定义也是常用的分数阶微积分定义之一. 相比 Caputo 型定义, 其对常数的分数阶导数不为零. 通常常数并不会影响一个函数的最优点, 而利用 Riemman-Liouville 型定义的算法最优点则会与函数的常数项相关, 影响算法的性能表现. 关于 Riemman-Liouville 型微积分相关优化算法的研究可以参考文献 [26].

对于优化问题:  $\min f(\theta)$ , 通过分数阶梯度的变初值策略, 截断二次及以上的高次项, 文献 [26] 得到了一类改进分数阶梯度下降算法, 具体可以表示为

$$\theta_{t+2} = \theta_{t+1} - \mu_t \frac{\nabla f(\theta_{t+1})}{\Gamma(2-\alpha)} (\theta_{t+1} - \theta_t)^{1-\alpha}, \quad (6)$$

其中  $0 < \alpha < 2$ ,  $\theta$  为待估参数,  $\mu_t$  为算法步长.

特别地, 对于集中式分数阶 LMS 算法的设计, 针对目标函数  $f(i) = e^2(i) = (d(i) - \mathbf{u}(i)\mathbf{w}(i-1))^2$ , 可以得到算法

$$\mathbf{w}(k+1) = \mathbf{w}(k) - \frac{\mu}{2} \frac{\partial^\alpha [e^2(k)]}{\partial \mathbf{w}^\alpha(k)}, \quad (7)$$

其中  $\mathbf{w}(k) \in \mathbb{R}^M$  是  $k$  时刻对参数的估计向量, 而  $\mu$  是算法的步长. 考虑向量  $\mathbf{w}$  的第  $j$  个分量, 可得

$$w_j(k+1) = w_j(k) - \frac{\mu}{2} \frac{\partial^\alpha [e^2(k)]}{\partial w_j^\alpha(k)}. \quad (8)$$

对于上面这样的迭代式, 文献 [31] 给出了改进后的一般分数阶梯度下降法:

$$w_j(k+1) = w_j(k) + \mu \frac{e(k)u(k-j+1)}{\Gamma(2-\alpha)} [w_j(k-1) - w_j(k-2)]^{1-\alpha}. \quad (9)$$

为了排除幂次项的奇异性, 通过添加一个小的扰动参数  $\delta$ , 可以得到集中式分数阶梯度 LMS 算法:

$$\mathbf{w}(k+1) = \mathbf{w}(k) + \mu \Gamma(\mathbf{w}, k-1, \alpha) e(k) \mathbf{u}^T(k), \quad (10)$$

其中

$$\Gamma(\mathbf{w}, k-1, \alpha) = \text{diag} \left\{ \frac{(|w_1(k-1) - w_1(k-2)| + \delta)^{1-\alpha}}{\Gamma(2-\alpha)}, \dots, \frac{(|w_M(k-1) - w_M(k-2)| + \delta)^{1-\alpha}}{\Gamma(2-\alpha)} \right\}.$$

### 2.2.1 固定阶次分数阶扩散 LMS 算法

受上述文献启发, 针对问题 (2), 本文为每个节点设计如下的分数阶 LMS 算法:

$$\mathbf{w}_{k,i} = \mathbf{w}_{k,i-1} + \mu_k \mathbf{w}_{k,i-2} \mathcal{D}_{\mathbf{w}_{k,i-1}}^\alpha J_k^g(\mathbf{w}_{k,i-1}), \quad (11)$$

其中  $\mathbf{w}_{k,i}$  是节点  $k$  在  $i$  时刻对参数的估计值,  $\mu_k$  为节点  $k$  的固定步长,  $\mathbf{w}_{k,i-2} \mathcal{D}_{\mathbf{w}_{k,i-1}}^\alpha J_k^g$  表示对目标函数的变初值分数阶导数, 该分数阶导数积分初值与终值均为时变参数. 我们对分数阶导数进行截断, 可得

$$\mathbf{w}_{k,i-2} \mathcal{D}_{\mathbf{w}_{k,i-1}}^\alpha J_k^g(\mathbf{w}_{k,i-1}) \approx \Gamma_{k,i} \left[ e_k(i) \mathbf{u}_{k,i}^T + \sum_{l \in \mathcal{N}_{k/k}} b_{lk} \|\mathbf{w}_{l,i-1} - \mathbf{w}^0\| \right], \quad (12)$$

**Algorithm 1** 固定阶次分数阶扩散 LMS 算法

输入:  $\mu_k > 0, \alpha_k, \mathbf{w}_{k,0}, \mathbf{w}_{k,1}$ ;  
 1: **for** 对于每个节点  $k = 1, \dots, N$  **do**  
 2:   **for** 对于样本数据点  $i = 1, \dots, n$  **do**  
 3:     适应过程, 更新本地估计参数:  $\phi_{k,i} = \mathbf{w}_{k,i-1} + \mu_k \Gamma_k(d_k(i)) \mathbf{u}_{k,i}^T - \mathbf{R}_{k,i} \mathbf{w}_{k,i-1}$ ;  
 4:     组合过程, 结合邻居信息更新参数:  $\mathbf{w}_{k,i} = \sum_{l \in \mathcal{N}_k} a_{lk} \phi_{l,i}$ ;  
 5:   **end for**  
 6: **end for**  
 输出:  $\mathbf{w}(n) = \frac{1}{N} \sum_{i=1}^N \mathbf{w}_{i,n}$ .

其中

$$\Gamma_{k,i} = \text{diag} \left\{ \frac{(|w_{k,1}(i-1) - w_{k,1}(i-2)| + \delta)^{1-\alpha}}{\Gamma(2-\alpha)}, \dots, \frac{(|w_{k,M}(i-1) - w_{k,M}(i-2)| + \delta)^{1-\alpha}}{\Gamma(2-\alpha)} \right\}.$$

采用 ATC 策略, 基于分数阶梯度的扩散 LMS 算法可以表述为

$$\phi_{k,i} = \mathbf{w}_{k,i-1} + \mu_k \Gamma_k(d_k(i)) \mathbf{u}_{k,i}^T - \mathbf{R}_{k,i} \mathbf{w}_{k,i-1}, \quad (13)$$

$$\mathbf{w}_{k,i} = \sum_{l \in \mathcal{N}_k} a_{lk} \phi_{l,i}, \quad (14)$$

其中  $\mathbf{R}_{k,i} = \mathbf{u}_{k,i}^T \mathbf{u}_{k,i}$ , 而权重矩阵  $\mathbf{A} = [a_{lk}]$  是行随机的, 即

$$a_{lk} \geq 0, \sum_{l=1}^N a_{lk} = 1; a_{kl} = 0 \text{ if } l \notin \mathcal{N}_k. \quad (15)$$

关于相邻节点之间的权重系数矩阵, 有多种取法<sup>[3]</sup>. 本文基于以下的权重矩阵来实现, 在切换拓扑的情况下,  $a_{l,k}$  也变成相应的  $a_{l,k}(t)$ , 其中  $n_k$  代表节点  $k$  的入度:

$$a_{lk} = \begin{cases} 1/\max\{n_k, n_l\}, & l \neq k, \\ 1 - \sum_{l \in \mathcal{N}_k/k} a_{lk}, & l = k. \end{cases} \quad (16)$$

这样, 我们就得到了固定阶次的分数阶扩散 LMS 算法 (见算法 1).

**注释 2** ATC 和 CTA 策略是最常用的两种扩散策略, 其主要区别在于节点在扩散网络中遇到新信息时行动顺序上有所不同. 从节点行为来看, ATC (适应 - 然后 - 组合) 策略中个体更注重自身的行为, 而 CTA (组合 - 然后 - 适应) 策略更注重网络的整合考虑. 从网络效应来看, ATC 策略可以使调整后信息更相关或可接受的个体策略更快地被采纳或扩散. 而 CTA 中的决策过程则会更加缓慢, 但是节点间的一致性更高. 从数学来看, ATC 和 CTA 策略的整体网络误差不相伯仲, 但是 ATC 策略在节点个体误差方面要优于 CTA 策略<sup>[14]</sup>. 在本文设计的变阶次分数阶扩散 LMS 算法中, 我们更加注重节点从自身信息出发, 更新策略, 充分发挥阶次对于节点的定制性, 故采用 ATC 策略作为基础策略.

### 2.2.2 变阶次分数阶扩散 LMS 算法

为了增加算法设计的自由度和修正固定步长带来的缺陷, 我们将算法 1 中的固定分数阶阶次调整

为变阶次. 本文设计如下的变分数阶梯度策略:

$$\alpha_k(i) = \begin{cases} \max\{\alpha_{k,m}, 0.1\}, & \alpha'_k(i) > \max\{\alpha_{k,m}, 0.1\}, \\ 0.1, & \alpha'_k(i) < 0.1, \\ \alpha'_k(i), & \text{否则}, \end{cases} \quad (17)$$

其中,

$$\alpha'_k(i) = \gamma\alpha_k(i-1) + \lambda e_k^2(i), \quad (18)$$

或者选择误差也在分数阶阶次之间转播,

$$\alpha'_k(i) = \gamma\alpha_k(i-1) + \lambda \sum_{l \in \mathcal{N}_k} a_{kl} e_l^2(i), \quad (19)$$

系数  $0 < \gamma < 1$  和  $\lambda > 0$  是人工设置的常数;

$$\alpha_{k,m} = \max \left\{ \alpha \mid \Gamma_{\max}(k, i) < \frac{1}{\mu_k \lambda(\mathbf{R}_{k,i})} \right\}, \quad (20)$$

$\Gamma_{\max}(k, i)$  是对角矩阵  $\mathbf{E}[\mathbf{\Gamma}_{k,i}] \approx \mathbf{\Gamma}_{k,i}$  的最大分量,  $\lambda(\mathbf{E}[\mathbf{R}_{k,i}])$  是  $\mathbf{E}[\mathbf{R}_{k,i}] \approx \mathbf{R}_{k,i} = \mathbf{u}_{k,i}^T \mathbf{u}_{k,i}$  的最大特征值. 在变分数阶扩散式 LMS 算法中, 相应的矩阵  $\mathbf{\Gamma}$  就变为

$$\mathbf{\Gamma}_{k,i} = \text{diag} \left\{ \frac{(|w_{k,1}(i-1) - w_{k,1}(i-2)| + \delta)^{1-\alpha_k(i)}}{\Gamma(2-\alpha_k(i))}, \dots, \frac{(|w_{k,M}(i-1) - w_{k,M}(i-2)| + \delta)^{1-\alpha_k(i)}}{\Gamma(2-\alpha_k(i))} \right\}. \quad (21)$$

**注释3** 分数阶算子的记忆特性主要体现在  $\mathbf{\Gamma}_{k,i}$  矩阵上, 通过对前一步信息的记录, 优化整个算法的迭代过程. 在面对无规则的系统噪声时, 通过历史信息的使用可以降低噪声的影响. 例如, 对于突然出现的脉冲噪声带来较大的  $e_k(i)$ , 由于矩阵  $\mathbf{\Gamma}_{k,i}$  记录的前一步信息没有脉冲噪声的影响, 在进行迭代时较大的噪声的算法波动也会变小.

**注释4** 在采用固定阶次的分数阶扩散 LMS 算法中, 为所有节点设计统一可靠的阶次是难以做到的, 这也与分布式优化的思想背道而驰, 完全丧失了分数阶算子定制性的优势. 为了解决此难点, 我们引入变阶次策略. 在改进分数阶 LMS 算法的形式中, 分数阶阶次直接影响算法的步长项, 与变步长 LMS 算法的思想类似, 算法初期较大的步长可以带来算法较快的收敛速度, 而算法稳态后较小的步长可以控制算法的稳态误差. 我们对变阶次的策略进行这样的设计, 也希望算法同时满足这两个要求, 同时从后续的收敛性分析中, 我们还为阶次设计了上下界, 以满足后续理论分析的需要. 变分数阶阶次算法优于变步长算法的方面在于分数阶直接以指数形式影响性能, 对算法的影响直观且有效.

**注释5** 显然  $\alpha_k(i)$  会通过矩阵  $\mathbf{\Gamma}_{k,i}$  来影响整体算法的瞬态和稳态性质. 在算法的执行过程中, 当迭代点不同分量  $j$  之间的距离  $|w_j(i-1) - w_k(i-2)|$  较大或估计误差  $e_k(i)$  较大时, 较大的阶次可以提高算法的收敛速度, 在算法趋于稳定时, 较小的阶次可以降低算法的稳态误差.

**引理1** ([27]) 对于任意正实数  $x < 0.5$ , 函数  $f(\alpha) = \frac{x^{1-\alpha}}{\Gamma(2-\alpha)}$  是一个关于自变量  $\alpha$  的单调递增函数, 其中  $0 < \alpha < 1.5$ .

**注释6** 当设计分数阶阶次 (20) 时, 其设计的上界与下界后面会用作收敛性证明. 其中上界  $\alpha_{k,m}$  是一个非线性函数的极值问题. 由引理 1 得, 函数  $f(\alpha) = \frac{x^{1-\alpha}}{\Gamma(2-\alpha)}$  在  $x < 0.5$  时存在单调性, 我们可以通过试错的方式找到一个  $\alpha_k(i)$ , 使得其满足  $\Gamma_{\max}(k, i) < \frac{1}{\mu_k \lambda(\mathbf{R}_{k,i})}$  即可, 而不必贴近其临界值. 而其下界 0.1 主要是作为一个算法进入稳态阶段的判断. 同时, 由于伽马函数  $\Gamma(2-\alpha)$  在  $\alpha \rightarrow 2$  时趋于无穷,  $\mathbf{\Gamma}_{k,i} \rightarrow \mathbf{0}$ , 故一般在实际应用中设置最大阶次不超过 1.5.

**Algorithm 2** 变阶次分数阶扩散 LMS 算法

---

输入:  $\mu_k > 0, \alpha_k(0) = 1.5, \lambda > 0, \gamma > 0, \mathbf{w}_{k,0}, \mathbf{w}_{k,1}$ ;  
 1: **for** 对于每个节点  $k = 1, \dots, N$  **do**  
 2:   **for** 对于样本数据点  $i = 1, \dots, n$  **do**  
 3:     更新分数阶阶次  $\alpha_k(i)$ ;  
 4:     适应过程, 更新本地估计参数:  $\phi_{k,i} = \mathbf{w}_{k,i-1} + \mu_k \mathbf{\Gamma}_k(d_k(i) \mathbf{u}_{k,i}^T - \mathbf{R}_{k,i} \mathbf{w}_{k,i-1})$ ;  
 5:     组合过程, 结合邻居信息更新参数:  $\mathbf{w}_{k,i} = \sum_{l \in \mathcal{N}_k} a_{kl} \phi_{l,i}$ ;  
 6:   **end for**  
 7: **end for**  
 输出:  $\mathbf{w}(n) = \frac{1}{N} \sum_{i=1}^N \mathbf{w}_{i,n}$ .

---

这样, 我们可以得到变阶次分数阶扩散 LMS 算法, 见算法 2.

本文主要分析算法 2 的收敛性能, 关于算法 1 的收敛性将作为推论给出.

### 3 性能分析

#### 3.1 固定拓扑下分数阶扩散 LMS 算法

本节主要分析固定拓扑下所设计算法的收敛性能和稳态性质. 在分析算法的相应性质之前, 我们需要如下假设.

**假设 1** (1) 所有的回归向量  $\mathbf{u}_{k,i}$  均为零均值, 且是时间、空间相互独立的.

(2) 任意节点  $k$  处的噪声信号  $v_{k,i}$  为一个高斯白噪声信号, 且与其他节点噪声信号独立.

**定理 1** 对于变分数阶扩散 LMS 算法 (算法 2), 在设计分数阶机制 (17) 的调节下, 各节点在固定迭代步长  $\mu_k$  下的估计向量  $\mathbf{w}_k, k = 1, \dots, N$  可以在统计意义下收敛至最优值  $\mathbf{w}^0$ . 此外, 越大的分数阶阶次  $\alpha_k(i) \in (0, 1.5)$  可以为估计向量在算法初期带来越快的收敛速度.

**证明** 先定义网络中各个节点的误差向量如下:

$$\tilde{\phi}_{k,i} = \mathbf{w}^0 - \phi_{k,i}, \quad \tilde{\mathbf{w}}_{k,i} = \mathbf{w}^0 - \mathbf{w}_{k,i}.$$

基于算法 2, 各个误差向量关于时间  $i$  的递推关系可以表示为

$$\tilde{\phi}_{k,i} = (\mathbf{I}_M - \mu_k \mathbf{\Gamma}_{k,i} \mathbf{R}_{k,i}) \tilde{\mathbf{w}}_{k,i-1} - \mu_k \mathbf{\Gamma}_{k,i} \mathbf{u}_{k,i}^T v_k(i), \quad (22)$$

$$\tilde{\mathbf{w}}_{k,i} = \sum_{l \in \mathcal{N}_k} a_{lk} \tilde{\phi}_{l,i}. \quad (23)$$

对于网络中所有节点的误差向量, 我们通过克罗内克积给出下列定义:

$$\mathbf{\Gamma} = \mu_k \mathbf{\Gamma}_{k,i} \otimes \mathbf{I}_N, \quad (24)$$

$$\tilde{\phi}_i = [\tilde{\phi}_{1,i}, \tilde{\phi}_{1,i}, \dots, \tilde{\phi}_{N,i}]^T, \quad \tilde{\mathbf{w}}_i = [\tilde{w}_{1,i}, \tilde{w}_{2,i}, \dots, \tilde{w}_{N,i}]^T.$$

类似地, 定义回归矩阵和权重矩阵:

$$\mathbf{R}_i = \text{diag}\{\mathbf{R}_{1,i}, \dots, \mathbf{R}_{N,i}\}, \quad (25)$$

$$\mathbf{A} = \mathbf{A} \otimes \mathbf{I}_M. \quad (26)$$

记

$$\theta_i = \text{col}\{\mathbf{u}_{1,i}^T v_1(i), \dots, \mathbf{u}_{N,i}^T v_N(i)\}, \quad (27)$$

则式 (22) 和 (23) 可以表示为

$$\tilde{\mathbf{w}}_i = \mathcal{A}^T(\mathbf{I}_{NM} - \mathbf{\Gamma}\mathbf{R}_i)\tilde{\mathbf{w}}_{i-1} - \mathcal{A}^T\mathbf{\Gamma}\theta_i, \quad (28)$$

在上式的两边同时取数学期望后, 有下式成立:

$$\mathbf{E}[\tilde{\mathbf{w}}_i] = \mathcal{A}^T(\mathbf{I}_{NM} - \mathbf{E}[\mathbf{\Gamma}\mathbf{R}_i])\mathbf{E}[\tilde{\mathbf{w}}_{i-1}]. \quad (29)$$

**注释7** 系统 (29) 的稳定性条件可以表述为  $|\lambda_{\max}(\mathcal{A}^T(\mathbf{I}_{NM} - \mathbf{E}[\mathbf{\Gamma}\mathbf{R}_i]))| < 1$ , 而  $\lambda_{\max}(\mathcal{A}^T(\mathbf{I}_{NM} - M\mathbf{R}_i)) \leq \|\mathcal{A}^T\|_2 \cdot |\lambda_{\max}(\mathbf{I}_{NM} - M\mathbf{R}_i)|$ , 因此稳定性条件转变为  $0 < \mu_k \mathbf{\Gamma}_{k,i} \mathbf{R}_{k,i} < 1$ . 这说明通过设计的变阶次策略, 收敛性条件是可以达到的.

**注释8** 由引理 1 可知, 较大的分数阶阶次  $\alpha$  可以带来较大的步长项  $\mathbf{\Gamma}$ , 从而提升算法初期的收敛速度, 而在算法接近或达到稳定时, 同时估计向量也满足引理 1 的要求, 此时较小的分数阶阶次可以减少算法的稳态误差, 即稳态误差与阶次之间满足单调性.

**定理2** 对于变分数阶扩散 LMS 算法 (算法 2), 在设计分数阶机制 (17) 的调节下, 各节点处的分数阶阶次  $\alpha_k(i) \in (0, 1.5)$  越小, 各节点处的估计向量的稳态误差也就越小, 同时网络中整体的估计向量的稳态误差也就越小.

**证明** 首先, 将各个节点的均方误差记作

$$\text{MSD}_{k,i} = \mathbf{E}[\|\tilde{\mathbf{w}}_{k,i}\|^2]. \quad (30)$$

故  $i$  时刻网络整体的均方误差可以表示为

$$\text{MSD}_i = \mathbf{E}[\|\tilde{\mathbf{w}}_i\|^2] = \frac{1}{N} \sum_{k=1}^N \text{MSD}_{k,i}. \quad (31)$$

令  $\gamma_i = \mathcal{A}^T(\mathbf{I}_{NM} - \mathbf{\Gamma}\mathbf{R}_i)$  和  $\eta = \mathcal{A}^T\mathbf{\Gamma}$ , 那么式 (28) 可以表述为

$$\tilde{\mathbf{w}}_i = \gamma_i \tilde{\mathbf{w}}_{i-1} - \eta \theta_i. \quad (32)$$

对于任意给定的 Hermite 矩阵  $\Sigma$ , 定义  $\|x\|_{\Sigma}^2 = x^T \Sigma x$ , 由式 (28) 可得

$$\|\tilde{\mathbf{w}}_i\|_{\Sigma}^2 = \|\gamma_i \tilde{\mathbf{w}}_{i-1} - \eta \theta_i\|_{\Sigma}^2 = \tilde{\mathbf{w}}_{i-1}^T \gamma_i^T \Sigma \gamma_i \tilde{\mathbf{w}}_{i-1} + \theta_i^T \eta^T \Sigma \eta \theta_i - \tilde{\mathbf{w}}_{i-1}^T \gamma_i^T \Sigma \eta \theta_i - \theta_i^T \eta^T \Sigma \gamma_i \tilde{\mathbf{w}}_{i-1}. \quad (33)$$

在上式两边同时取期望, 可得

$$\mathbf{E}[\|\tilde{\mathbf{w}}_i\|_{\Sigma}^2] = \mathbf{E}[\tilde{\mathbf{w}}_{i-1}^T \gamma_i^T \Sigma \gamma_i \tilde{\mathbf{w}}_{i-1}] + \mathbf{E}[\theta_i^T \eta^T \Sigma \eta \theta_i]. \quad (34)$$

利用  $\text{Tr}(\mathbf{A}\mathbf{B}) = \text{Tr}(\mathbf{B}\mathbf{A})$ , 上式的第二项可以写为

$$\mathbf{E}[\theta_i^T \eta^T \Sigma \eta \theta_i] = \text{Tr}[\eta^T \Sigma \eta \mathbf{E}[\theta_i \theta_i^T]] = \text{Tr}[\Sigma \eta \Theta \eta^T],$$

其中  $\Theta = \mathbf{E}[\theta_i \theta_i^T]$ , 同时式 (34) 的第一项可以表示为

$$\mathbf{E}[\tilde{\mathbf{w}}_{i-1}^T \gamma_i^T \Sigma \gamma_i \tilde{\mathbf{w}}_{i-1}] = \mathbf{E}[\mathbf{E}[\tilde{\mathbf{w}}_{i-1}^T \gamma_i^T \Sigma \gamma_i \tilde{\mathbf{w}}_{i-1} | \tilde{\mathbf{w}}_{i-1}]] = \mathbf{E}[\tilde{\mathbf{w}}_{i-1}^T \Sigma' \tilde{\mathbf{w}}_{i-1}] = \mathbf{E}[\|\tilde{\mathbf{w}}_{i-1}\|_{\Sigma'}^2],$$



其中  $\Sigma' = E[\gamma_i^T \Sigma \gamma_i]$ .

记

$$E[\gamma_i] = \mathcal{A}^T (\mathbf{I}_{NM} - M \mathbf{R}_i) = \gamma, \quad (35)$$

以及  $\sigma = \text{vec}(\Sigma)$ ,  $\sigma' = \text{vec}(\Sigma')$ , 且注意到下列关系式:

$$\text{vec}(\mathbf{U} \Sigma \mathbf{W}) = (\mathbf{W}^T \otimes \mathbf{U}) \sigma, \quad \text{Tr}(\Sigma \mathbf{W}) = [\text{vec}(\mathbf{W}^T)]^T \sigma.$$

通过这些变形并略去步长的二次项, 可以得到

$$\begin{aligned} \sigma' &= \text{vec}(\mathcal{A} \Sigma \mathcal{A}^T) - \text{vec}(\mathcal{A} \Sigma \mathcal{A}^T \Gamma \mathbf{R}_i) - \text{vec}(\mathbf{R}_i^T \Gamma \mathcal{A} \Sigma \mathcal{A}^T) \\ &= (\mathcal{A} \otimes \mathcal{A}) \sigma - [\mathbf{R}_i^T \Gamma \mathcal{A} \otimes \mathcal{A}] \sigma - [\mathcal{A} \otimes \mathbf{R}_i^T \Gamma \mathcal{A}] \sigma = \mathcal{F} \sigma, \end{aligned}$$

其中  $\mathcal{F} = \gamma^T \otimes \gamma \approx (\mathcal{A} \otimes \mathcal{A}) - [\mathbf{R}_i^T \Gamma \mathcal{A} \otimes \mathcal{A}] - [\mathcal{A} \otimes \mathbf{R}_i^T \Gamma \mathcal{A}]$ .

最后,  $\tilde{w}_i$  和  $\tilde{w}_{i-1}$  之间的二阶矩关系可以表示为

$$E[\|\tilde{w}_i\|_{\Sigma}^2] = E[\tilde{w}_{i-1}^T \gamma_i^T \Sigma \gamma_i \tilde{w}_{i-1}] + E[\theta_i^T \eta^T \Sigma \eta \theta_i] = E[\|\tilde{w}_{i-1}\|_{\Sigma'}^2] + [\text{vec}(\eta \Theta \eta^T)]^T \sigma.$$

根据  $\text{vec}(\|\tilde{w}_i\|_{\Sigma}^2) = (\tilde{w}_i^T \otimes \tilde{w}_i) \sigma$ , 上式也可以写为

$$E[\|\tilde{w}_i\|_{\Sigma}^2] = E[\|\tilde{w}_{i-1}\|_{\mathcal{F}\sigma}^2] + [\text{vec}(\eta \Theta \eta^T)]^T \sigma. \quad (36)$$

从而当算法进入稳态环节时, 即  $i \rightarrow \infty$ , 有  $\|\tilde{w}_i\| \approx \|\tilde{w}_{i-1}\|$ , 故

$$E[\|\tilde{w}_{\infty}\|_{(\mathbf{I}-\mathcal{F})\sigma}^2] = [\text{vec}(\eta \Theta \eta^T)]^T \sigma. \quad (37)$$

根据设计的变分数阶的上界, 可以保证矩阵  $\mathcal{F} = \gamma^T \otimes \gamma \approx E[\mathcal{A}^T (\mathbf{I}_{NM} - \Gamma \mathbf{R}_i)] \otimes E[\mathcal{A}^T (\mathbf{I}_{NM} - \Gamma \mathbf{R}_i)]$  是稳定的. 根据引理 1 知, 越小分数阶阶次  $\alpha$ , 可以带来越小的  $\eta$ , 从而降低算法的稳态误差.

**推论 1** 在固定阶次的分数阶扩散 LMS 算法 (算法 1) 中, 如果各节点处采用固定步长  $\mu_k$ ,  $k = 1, \dots, N$ , 由式 (29) 可知, 收敛性条件  $\mu_k < \frac{1}{\Gamma_{\max(k,i)} \lambda(\mathbf{R}_{k,i})}$  不能保证, 即算法后期的收敛性能会有影响.

**注释 9** 与算法 2 相比,  $E[\tilde{w}_i] = \mathcal{A}^T (\mathbf{I}_{NM} - E[\Gamma \mathbf{R}_i]) E[\tilde{w}_{i-1}]$  中的  $E[\Gamma \mathbf{R}_i]$  与固定分数阶阶次  $\alpha$  的选择有关. 若使用较大的阶次, 可以在算法初期获得更快的收敛速度, 但由于缺乏阶次调节的能力,  $E[\|\tilde{w}_{\infty}\|_{(\mathbf{I}-\mathcal{F})\sigma}^2] = [\text{vec}(\eta \Theta \eta^T)]^T \sigma$  中的  $\eta = \mathcal{A}^T \Gamma$  也会较大, 从而损失了算法的一部分稳态误差.

### 3.2 切换拓扑下算法的收敛性

首先, 为了证明切换有向图上的分数阶扩散 LMS 算法的收敛性, 需要下面一些假设和引理.

**假设 2** 存在一个  $\hat{\alpha} \in (0, 1)$  使得对于任意满足  $a_{ij}(t) > 0$  的  $a_{ij}(t)$ , 都有  $a_{ij}(t) \geq \hat{\alpha}$ , 其中  $a_{ij}(t)$  为  $t$  时刻节点  $i$  和  $j$  之间的通信权重.

**假设 3** 在任意时刻, 网络中的每个节点都存在一个自环, 即  $\alpha_{ii}(t) > 0$ .

**假设 4** 对于任意时刻  $t > 0$ , 均存在一个自然数  $p$ , 使得图  $\mathcal{G}[tp, (t+1)p]$  都包含一个生成树, 其固定根节点为  $r$ ,  $1 \leq r \leq n$ .

**引理 2** ([15]) 当网络拓扑满足假设 2~4 时, 存在正整数  $M > 0$ , 使得对于任意节点  $i = 1, \dots, N$  和任意时刻  $t_2 - t_1 \geq M$ , 存在时间  $[t_1, t_2]$  上从根节点  $r$  到目标节点  $i$  的路径.

**注释10** 假设 2~4 对于切换拓扑下的算法是非常常见的假设. 假设 2 是指网络中的节点之间若存在连通路, 则连通权重不会趋于 0. 一方面由于节点数有限, 在我们采用的权重结构中不会出现这个问题. 另一方面实际应用中系统的通信路线如果趋于 0, 则认为这个通信不存在. 假设 3 指每个节点都可以获得自身的信息, 而假设 4 是指切换图在有限时间内至少有一次直接或间接的通信, 否则不能保证一致性的实现.

**定理3** 在算法扩散网络为确定的切换有向网络时, 如果其网络拓扑满足上述假设 2~4, 那么网络中的各节点的估计向量  $\mathbf{w}_k, k = 1, \dots, N$  可以在统计意义下收敛到最优向量  $\mathbf{w}^0$ .

**证明** 对于上述系统 (29), 此时, 系统转变为确定性的时变系统

$$\mathbf{E}[\tilde{\mathbf{w}}_i] = \mathcal{A}(i)(\mathbf{I}_{NM} - M\mathbf{R}_i)\mathbf{E}[\tilde{\mathbf{w}}_{i-1}]. \quad (38)$$

记

$$m(i) = \max_{k=1, \dots, N} \|\mathbf{E}[\tilde{\mathbf{w}}_{k,i}]\| \geq 0,$$

于是

$$\begin{aligned} m(i+1) &= \max_{k=1, \dots, N} \|\mathbf{E}[\tilde{\mathbf{w}}_{k,i+1}]\| = \max_{k=1, \dots, N} \sum_{j=1}^N a_{ij}(i) \|\mathbf{E}[\tilde{\mathbf{w}}_{j,i}]\| \\ &\leq m(i) \sum_{j=1}^N a_{ij}(i) \|\mathbf{I} - \mu_k \mathbf{\Gamma}_{k,i} \mathbf{R}_{k,i}\| \leq m(i) \left(1 - \mu_k \frac{\delta^{0.9}}{\Gamma(1.9)} \lambda_{\min}(\mathbf{R}_i)\right) < m(i). \end{aligned}$$

由于  $\mathbf{\Gamma}_{k,i}$  的设计中存在干扰参数  $\delta$  和最小分数阶 0.1, 故倒数第二个不等式成立. 所以  $m(i) \geq 0$  且  $m(i+1) \leq m(i)$ , 故  $m^* \triangleq \lim_{i \rightarrow \infty} m(i) = 0$ .

对于固定的时间  $\tau$ , 如果存在一条从节点  $j$  到  $i$  的边, 可以得到

$$\begin{aligned} \|\mathbf{E}[\tilde{\mathbf{w}}_{k,i+1}]\| &= \left\| \mathbf{E} \left[ \sum_{q=1}^N a_{iq}(i) \phi_{q,i+1} \right] \right\| \leq \sum_{q=1}^N a_{iq}(i) \|\mathbf{E}[\tilde{\mathbf{w}}_{q,i}]\| \\ &= \sum_{q \neq j}^N a_{iq}(i) \|\mathbf{E}[\tilde{\mathbf{w}}_{q,i}]\| + a_{ij}(i) \|\mathbf{E}[\tilde{\mathbf{w}}_{j,i}]\| \\ &\leq m(i) \sum_{q=1}^N a_{iq}(i) + a_{ij}(i) [\|\mathbf{E}[\tilde{\mathbf{w}}_{j,t}]\| - m(i)] \\ &\leq m(i) - \hat{\alpha} [m(i) - \|\mathbf{E}[\tilde{\mathbf{w}}_{j,t}]\|]. \end{aligned} \quad (39)$$

因此, 对于一个由节点  $i$  到节点  $j$  的动态路径序列  $E(t_1), E(t_1+1), \dots, E(t_2)$ , 其中  $E(t_1)$  是以节点  $i$  为起点的有向边,  $E(t_2)$  是以节点  $j$  为终点的有向边.

使用数学归纳法可以得到

$$m(t_1) - \|\mathbf{E}[\tilde{\mathbf{w}}_{i,t_2}]\| \geq \hat{\alpha}^{t_2-t_1} [m(t_1) - \|\mathbf{E}[\tilde{\mathbf{w}}_{j,t_1}]\|]. \quad (40)$$

具体过程为: 当  $t_1 = t_2$  时, 式 (45) 显然成立. 假定对于某个  $t_2 > t_1$ , 式 (40) 成立, 那么对于动态路径序列  $E(t_1), E(t_1+1), \dots, E(t_2+1)$ , 其中  $j_{-1}$  作为边  $E(t_2)$  的起点, 根据假设有  $m(t_1) - \|\mathbf{E}[\tilde{\mathbf{w}}_{j_{-1},t_2}]\| \geq \hat{\alpha}^{t_2-t_1} [m(t_1) - \|\mathbf{E}[\tilde{\mathbf{w}}_{j,t_1}]\|]$ , 注意到式 (39) 成立, 可得  $m(t_1) - \|\mathbf{E}[\tilde{\mathbf{w}}_{i,t_2+1}]\| \geq \hat{\alpha}^{t_2-t_1+1} [m(t_1) - \|\mathbf{E}[\tilde{\mathbf{w}}_{j,t_1}]\|]$ , 故得证.

另一方面, 由引理 2 可知, 对于任意  $\tau > 0$  和节点  $i$ , 均存在一路径由根节点  $r$  到达节点  $i$ . 故取  $t_1 = \tau, t_2 = \tau + M$ , 可以得到

$$m(\tau) - \|\mathbf{E}[\mathbf{w}_{i,\tau+M}]\| \geq \hat{\alpha}^M [m(\tau) - \|\mathbf{E}[\mathbf{w}_{r,\tau}]\|].$$

根据  $m(i)$  的定义, 有

$$m(\tau) - m(\tau + M) \geq \hat{\alpha}^M [m(\tau) - \|\mathbf{E}[\mathbf{w}_{r,\tau}]\|]. \quad (41)$$

记根节点处的误差函数为  $\epsilon(t) = m(t) - \|\mathbf{E}[\tilde{\mathbf{w}}_{r,t}]\|$ . 下证级数  $\{\epsilon(t)\}_{t=0}^{\infty}$  是可加的. 取任意整数  $\theta \geq 0$  和时间序列  $\{\tau = \theta + qM\}, q \geq 0$  为一个整数, 有

$$\begin{aligned} \sum_{q=0}^{\infty} \epsilon(\theta + qM) &= \sum_{q=0}^{\infty} [m(\theta + qM) - \|\mathbf{E}[\tilde{\mathbf{w}}_{r,\theta+qM}]\|] \\ &\leq \hat{\alpha}^{-M} \sum_{q=0}^{\infty} [m(\theta + qM) - m(\theta + (q+1)M)] \\ &= \hat{\alpha}^{-M} \left[ m(\theta) - \lim_{p \rightarrow \infty} m(\theta + (p+1)M) \right] \\ &= \hat{\alpha}^{-M} [m(\theta) - \hat{m}]. \end{aligned} \quad (42)$$

故

$$\sum_{t=0}^{\infty} \epsilon(t) = \sum_{\theta=0}^{M-1} \sum_{q=0}^{\infty} \epsilon(\theta + qM) \leq \hat{\alpha}^{-M} \left[ \sum_{\theta=0}^{M-1} m(\theta) - M\hat{m} \right] < \infty.$$

最后, 得到通项级数  $\epsilon(t) = m(t) - \|\mathbf{E}[\mathbf{w}_{r,t}]\| \rightarrow 0$ , 即  $\mathbf{w}_k \rightarrow \mathbf{w}_r \rightarrow \mathbf{w}_0$ . 这说明在切换拓扑中各节点也满足收敛性质.

**推论 2** 对于固定分数阶次的分数阶扩散 LMS 算法 (算法 1), 网络中的各节点的估计向量  $\mathbf{w}_k, k = 1, \dots, N$  在统计意义下收敛到最优向量  $\mathbf{w}^0$ .

## 4 数值仿真

为了验证所提算法的有效性和优越性, 本节将会提供算法的一些数值算例, 包括与一些相近算法例如 DLMS, VSSDLMS, QDLMS, NDLMS 的比较以及所提算法在系统识别中的应用. 其中所有的仿真结果都是 50 次独立实验后平均的结果, 仿真中参数的验证如图 2 所示. 本文所采用的指标是网络所有节点的平均偏差:

$$\overline{\text{MSD}} = \frac{1}{N} \sum_{k=1}^N \mathbf{E}[\|\tilde{\mathbf{w}}_{k,i}\|^2]. \quad (43)$$

### 4.1 固定拓扑上的数值仿真

采用的分布式网络中由 5 节点构成无向图, 其拓扑关系如图 3 所示.

在第一个算例中, 所要估计的目标向量  $\mathbf{w}^0 = [-0.5, -0.2]$ , 分数阶算法需要至少两步初始信息, 故不妨设  $\mathbf{w}_{k,0} = [-1, -1], \mathbf{w}_{k,1} = [-0.99, -0.99], \forall k$ , 参数设置为  $\mu = 0.1, \gamma = 0.8, \lambda = 0.01$ , 固定分数阶算法的阶次设置为  $\alpha = 1.2$ , 变步长算法的机制设置为  $\mu_k(i) = \gamma\mu_k(i-1) + \lambda e_k^2(i)$ . 同时我们对平稳

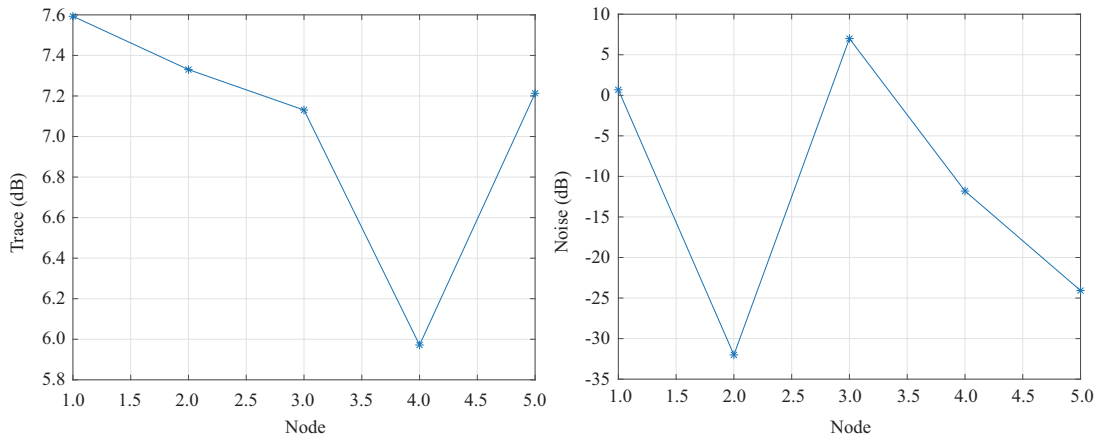


图 2 (网络版彩图) 数值仿真的参数  
 Figure 2 (Color online) Parameters in simulation

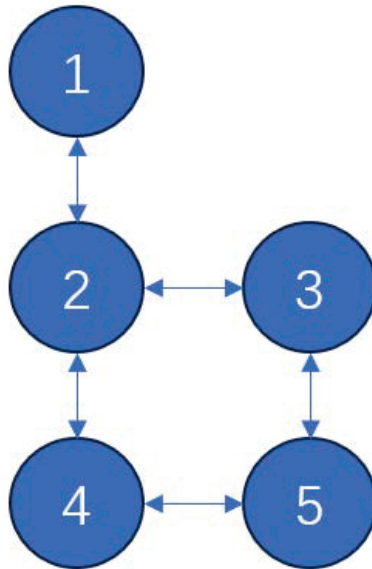


图 3 (网络版彩图) 网络的拓扑结构  
 Figure 3 (Color online) Topology of the network

和非平稳两种情况, 也就是  $\mathbf{u}_{k,i}$  是否满足假设 1 的情况, 分别进行了数值实验. 在攻击节点的情况下, 将节点 4 视为攻击节点, 攻击信号为  $w_{\text{attack}} = [0.2; 0.1]$ .

首先, 从两个不同阶次算法来验证对目标函数分数阶导数进行截断导致的算法性能差异. 因为 LMS 的目标函数仅由两项构成, 我们截断会损失目标函数的二阶导数部分. 从图 4 可以看出, 对于平稳环境下的算法, 二者的误差处于一个极小的范围内, 几乎可以忽略不计. 如果不进行截断的话, 算法计算量的提升并不会得到性能上的提升.

接下来验证不同的固定阶次以及变阶次分数阶扩散 LMS 算法的性能. 从图 5 中可以看出, 随着分数阶阶次从 1.2 减少到 0.8, 算法的收敛速度也在减小, 而变阶次算法 (varying fractional order diffusion least mean square, VFODLMS) 展示了收敛速度和稳态两方面的优势. 这一方面验证了上文叙述阶次

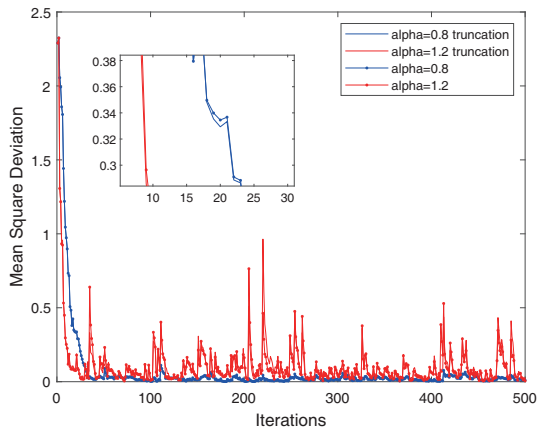


图 4 (网络版彩图) 平稳环境下截断导数引起的收敛性能差异 (MSD)

Figure 4 (Color online) Convergence performance of the algorithm caused by truncation

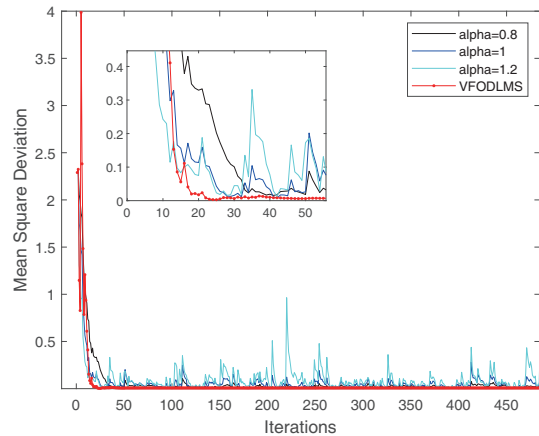


图 5 (网络版彩图) 平稳环境下各固定阶次与变阶次算法的收敛性能 (MSD)

Figure 5 (Color online) Convergence performance of the algorithm under fixed orders and variable order

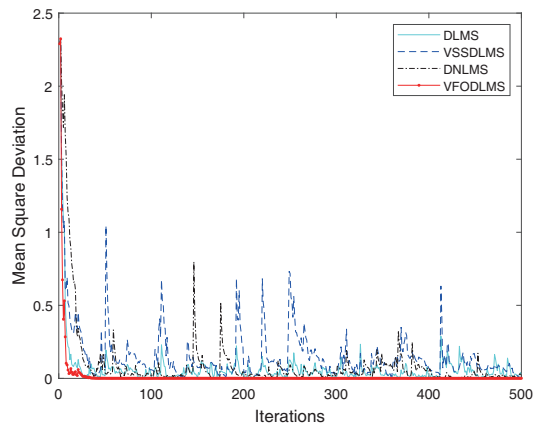
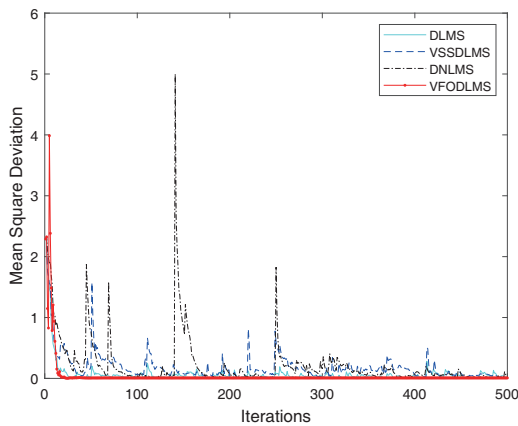


图 6 (网络版彩图) 无节点攻击下的平稳与非平稳环境下的算法收敛性能 (MSD)

Figure 6 (Color online) Convergence performance of the algorithm in the stationary and non-stationary environments without node attacks

与收敛速度的关系, 另一方面也展示了变阶次策略相对于固定阶次的优越性.

图 6 和 7 中显示的红色标记的曲线, 即本文设计的 VFODLMS 算法, 相比其他算法, 在收敛速度和稳态误差方面都表现出优越性. 由于特定数据点的影响, 相比正则扩散 LMS 算法 (diffusion normal least square, DNLMs) 和给定参数下的变参数 LMS 算法 (VSSDLMS), 本文设计的算法对于大范围的参数扰动在稳态误差方面表现出了相对稳定的性能. 同时在存在攻击节点的网络收敛性能图中, 攻击节点对于整体算法的影响几乎可以忽略.

#### 4.2 切换拓扑上的数值仿真

本小节采用确定性的切换拓扑, 具体拓扑结果如下, 满足假设 2~4.

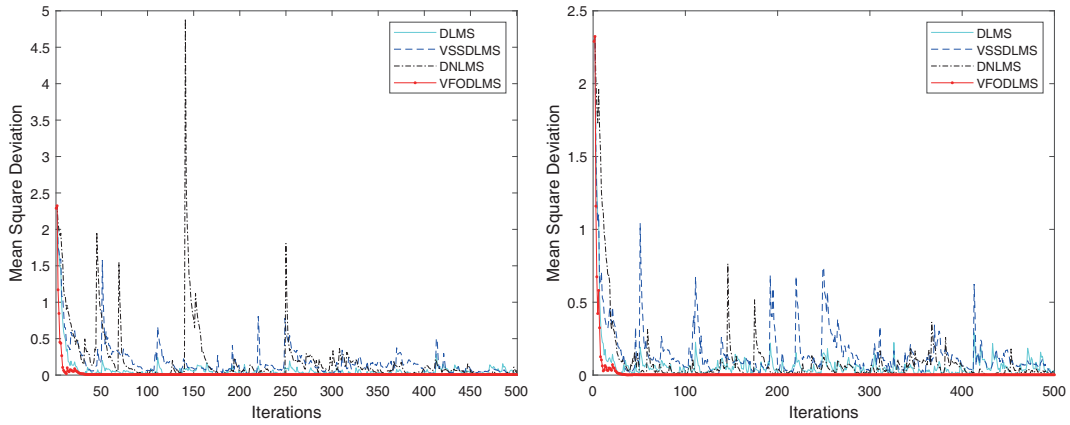


图 7 (网络版彩图) 存在节点攻击下的平稳与非平稳环境下的算法收敛性能 ( $\overline{\text{MSD}}$ )

Figure 7 (Color online) Convergence performance of the algorithm in the stationary and non-stationary environments under node attacks

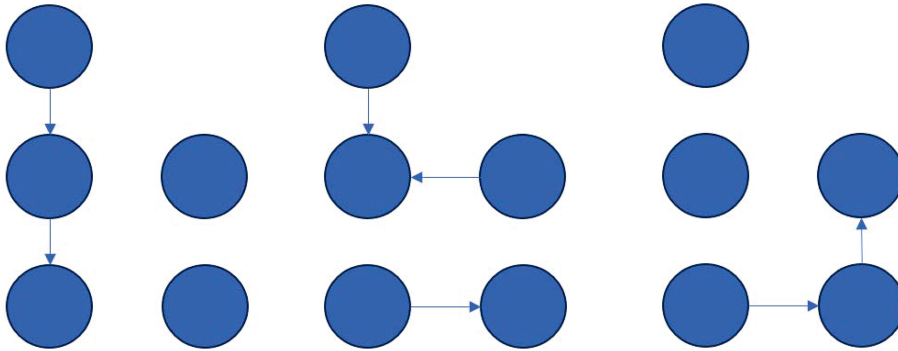


图 8 (网络版彩图) 切换拓扑的结构  $A_{1-3}$

Figure 8 (Color online) Structures of switching topologies

假设网络切换机制如下, 其拓扑结构由图 8 给出:

$$A(i) = \begin{cases} A_1, & i \bmod 3 = 0, \\ A_2, & i \bmod 3 = 1, \\ A_3, & \text{否则}. \end{cases} \quad (44)$$

从图 9 中可以看出, 本文提出的红色曲线表示的 VFODLMS 算法依然表现出了良好的收敛性能. 尤其在算法的稳态误差方面, 本文算法的稳态误差不受环境新给出的数据的影响, 保持了一个极其稳定的尺度. 同时相比于图 6 与 7 中采用的固定拓扑结构, 切换拓扑下的算法整体收敛速度也相对延迟大约 20 次迭代, 但是依然能够在 50 次迭代左右达到可接受的收敛效果.

## 5 结论

本文主要研究了分布式网络中的适应性参数估计问题. 利用阶次调节机制设计了一类基于分数阶梯度信息的变阶次扩散式 LMS 算法. 在固定步长下, 从理论角度分析并讨论了设计算法的收敛速度、

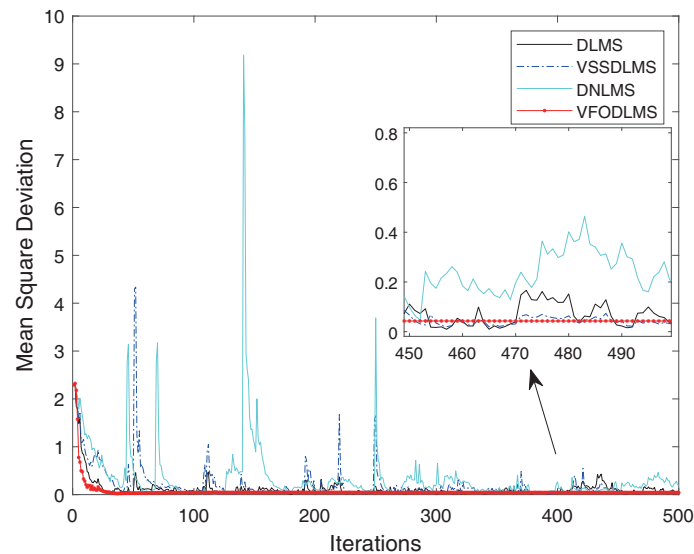


图9 (网络版彩图) 平稳环境下切换拓扑的算法收敛性能 ( $\overline{\text{MSD}}$ )

**Figure 9** (Color online) Convergence performance of the algorithm in stationary environments under the switched topologies

收敛精度等性质. 进一步, 给出了切换通信拓扑下算法的收敛条件. 最后, 针对在平稳与非平稳环境、是否存在攻击节点等情况, 从网络整体的平均偏差角度分析并验证了所提算法的有效性和优越性.

## 参考文献

- 1 Lopes C G, Sayed A H. Diffusion least-mean squares over adaptive networks: formulation and performance analysis. *IEEE Trans Signal Process*, 2008, 56: 3122–3136
- 2 Cattivelli F S, Sayed A H. Diffusion LMS strategies for distributed estimation. *IEEE Trans Signal Process*, 2009, 58: 1035–1048
- 3 Takahashi N, Yamada I, Sayed A H. Diffusion least-mean squares with adaptive combiners: formulation and performance analysis. *IEEE Trans Signal Process*, 2010, 58: 4795–4810
- 4 Ma L, Hu C, Yu J, et al. Distributed fixed/preassigned-time optimization based on piecewise power-law design. *IEEE Trans Cybern*, 2023, 53: 4320–4333
- 5 Estrin D, Girod L, Pottie G, et al. Instrumenting the world with wireless sensor networks. In: *Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing*, Salt Lake City, 2001. 2033–2036
- 6 Ghauri S A, Sohail M F. System identification using LMS, NLMS and RLS. In: *Proceedings of IEEE Student Conference on Research and Development*, Putrajaya, 2013. 65–69
- 7 Cattivelli F S, Sayed A H. Modeling bird flight formations using diffusion adaptation. *IEEE Trans Signal Process*, 2011, 59: 2038–2051
- 8 Sayed A H, Tu S Y, Chen J, et al. Diffusion strategies for adaptation and learning over networks: an examination of distributed strategies and network behavior. *IEEE Signal Process Mag*, 2013, 30: 155–171
- 9 Akyildiz I F, Weilian Su I F, Sankarasubramaniam Y, et al. A survey on sensor networks. *IEEE Commun Mag*, 2002, 40: 102–114
- 10 Degroot M H. Reaching a consensus. *J Am Stat Assoc*, 1974, 69: 118–121
- 11 Sayed A H. Adaptation, learning, and optimization over networks. *FNT Machine Learn*, 2014, 7: 311–801
- 12 Kar S, Moura J M F. Consensus + innovations distributed inference over networks: cooperation and sensing in networked systems. *IEEE Signal Process Mag*, 2013, 30: 99–109
- 13 Saeed M O B, Zerguine A, Hameed U, et al. An incremental noise constrained LMS algorithm. *Signal Process*, 2023,

- 213: 109187
- 14 Tu S Y, Sayed A H. Diffusion strategies outperform consensus strategies for distributed estimation over adaptive networks. *IEEE Trans Signal Process*, 2012, 60: 6217–6234
  - 15 Xie S Y, Guo L. Analysis of distributed adaptive filters based on diffusion strategies over sensor networks. *IEEE Trans Automat Contr*, 2018, 63: 3643–3658
  - 16 Fan L, Jia L J, Tao R, et al. Distributed bias-compensated normalized least-mean squares algorithms with noisy input. *Sci China Inf Sci*, 2018, 61: 112210
  - 17 Matsuo M V, Kuhn E V, Seara R. On the diffusion NLMS algorithm applied to adaptive networks: stochastic modeling and performance comparisons. *Digital Signal Process*, 2021, 113: 103018
  - 18 Zeng J, Mi W, Zheng W X. A diffusion bias-compensated LMS algorithm for distributed estimation with ARMAX models. *Digital Signal Process*, 2023, 142: 104202
  - 19 Arif M, Moinuddin M, Naseem I, et al. Diffusion quantum-least mean square algorithm with steady-state analysis. *Circ Syst Signal Process*, 2022, 41: 3306–3327
  - 20 Abadi M S E, Shafiee M S. Distributed estimation over an adaptive diffusion network based on the family of affine projection algorithms. *IEEE Trans Signal Inf Process Networks*, 2018, 5: 234–247
  - 21 Sayed A H. Diffusion adaptation over networks. *Acad Press Library Signal Process*, 2014, 3: 323–453
  - 22 Lee H S, Kim S E, Lee J W, et al. A variable step-size diffusion LMS algorithm for distributed estimation. *IEEE Trans Signal Process*, 2015, 63: 1808–1820
  - 23 Abdolee R, Vakilian V, Champagne B. Tracking performance and optimal adaptation step-sizes of diffusion-LMS networks. *IEEE Trans Control Netw Syst*, 2016, 5: 67–78
  - 24 Qadri S S U, Arif M, Saeed M O B. A novel variable step-size LMS algorithm for decentralized incremental distributed networks. *Circ Syst Signal Process*, 2023, 42: 7226–7249
  - 25 Sun Y, Hu C, Yu J. Bipartite leaderless synchronization of fractional-order coupled neural networks via edge-based adaptive pinning control. *J Franklin Inst*, 2024, 361: 1303–1317
  - 26 Chen Y, Gao Q, Wei Y, et al. Study on fractional order gradient methods. *Appl Math Comput*, 2017, 314: 310–321
  - 27 Cheng S, Wei Y, Chen Y, et al. An innovative fractional order LMS based on variable initial value and gradient order. *Signal Process*, 2017, 133: 260–269
  - 28 Yang C, Yang Z, Liao S, et al. Triple-GAN with variable fractional order gradient descent method and mish activation function. In: *Proceedings of the 12th International Conference on Intelligent Human-Machine Systems and Cybernetics (IHMSC)*, Hangzhou, 2020. 244–247
  - 29 Chen W, Wen C, Hua S, et al. Distributed cooperative adaptive identification and control for a group of continuous-time systems with a cooperative PE condition via consensus. *IEEE Trans Automat Contr*, 2013, 59: 91–106
  - 30 Wei Y, Chen Y Q, Gao Q, et al. Infinite series representation of functions in fractional calculus. In: *Proceedings of Chinese Automation Congress (CAC)*, Hangzhou, 2019. 1697–1702
  - 31 Zhang H, Mo L. An improved LMS algorithm based on fractional order gradient direction. In: *Proceedings of the 41st Chinese Control Conference (CCC)*, Hefei, 2022. 3156–3161



# A class of diffusion LMS algorithm with variable fractional order gradient

Yang YANG<sup>1</sup>, Lipo MO<sup>1,2\*</sup>, Min ZUO<sup>2</sup> & Yongguang YU<sup>3</sup>

1. *School of Mathematics and Statistics, Beijing Technology and Business University, Beijing 100048, China;*

2. *School of Computer and Artificial Intelligence, Beijing Technology and Business University, Beijing 100048, China;*

3. *School of Mathematics and Statistics, Beijing Jiaotong University, Beijing 100044, China*

\* Corresponding author. E-mail: molipo@th.btbu.edu.cn

**Abstract** For the parameter estimation issues in distributed networks, this paper mainly proposes a class of diffusion least mean squares (LMS) algorithm. A variable mechanism of fractional gradient orders is introduced to enhance the performance of the proposed algorithm. First, we extend the centralized fractional gradient LMS algorithms to the parameter estimation problems in distributed networks and study the convergence speed and accuracy of the proposed algorithm. Second, considering the effect of fractional orders on algorithm performance, a strategy with variable fractional orders is introduced to enhance the convergence performance of the proposed algorithm by giving full play to the advantage of fractional orders. Besides, we also prove the convergence of the proposed algorithm under switched topologies. Finally, some numerical simulation results are provided to validate the effectiveness and superiority of the proposed algorithm from the aspects of convergence speed, accuracy, and robustness.

**Keywords** distributed estimation, diffusion LMS algorithm, adaptive filter, switching topologies, fractional calculus