

# 扩散式变阶数最大相关熵准则算法

林云 黄桢航 高凡

重庆邮电大学通信与信息工程学院 重庆 400065

(lycqupt@sina.com)



**摘要** 固定阶数的分布式自适应滤波算法只有在待估计向量的阶数已知且恒定的情况下才能达到相应的估计精度,在阶数未知或时变的情况下算法的收敛性能会受到影响,变阶数的分布式自适应滤波算法是解决上述问题的有效途径。但是目前大多数分布式变阶数自适应滤波算法以最小均方误差(Mean square Error, MSE)准则作为滤波器阶数的代价函数,在脉冲噪声环境下算法的收敛过程会受到较大影响。最大相关熵准则具有对脉冲噪声的强鲁棒性,且计算复杂度低。为提高分布式变阶数自适应滤波算法在脉冲噪声环境下的估计精度,利用最大相关熵准则作为滤波器阶数迭代的代价函数,并将得到的结果代入固定阶数的扩散式最大相关熵准则算法,提出了一种扩散式变阶数最大相关熵准则(Diffusion Variable Tap-length Maximum Correntropy Criterion, DVTMCC)算法。通过与邻域的节点进行通信,所提算法以扩散的方式实现了整个网络的信息融合,具有估计精度高、计算量小等优点。仿真实验对比了在脉冲噪声下 DVTMCC 算法和其他分布式变阶数自适应滤波算法、固定阶数的扩散式最大相关熵准则算法的收敛性能。仿真结果表明,在脉冲噪声环境下 DVTMCC 算法能够同时估计未知向量的阶数和权值,性能优于参与对比的算法。

**关键词:** 自适应网络;扩散式策略;最大相关熵准则;变阶数;脉冲噪声

**中图分类号** TN911.72

## Diffusion Variable Tap-length Maximum Correntropy Criterion Algorithm

LIN Yun, HUANG Zhen-hang and GAO Fan

Department of Communication and Information Engineering, Chongqing University of Posts and Telecommunications, Chongqing 400065, China

**Abstract** The fixed tap-length distributed adaptive filtering algorithm can achieve the corresponding estimation accuracy only when the tap-length of the unknown vector is assumed to be known as a prior and constant. The convergence performance of the algorithm deteriorates when the tap-length is unknown or time varying. Variable tap-length distributed adaptive filtering algorithm is an effective way to solve this problem. However, most of the distributed variable tap-length adaptive filtering algorithms use the minimum mean square error (MSE) criterion as the cost function of the tap-length, and the convergence of the algorithm is greatly affected under the impulsive noise environment. The maximum correntropy criterion is robust to impulse noise and has low computational complexity. In order to improve the estimation accuracy of the distributed variable tap-length adaptive filtering algorithm under the impulsive noise environment, the maximum correntropy criterion is used as the cost function, relevant results are substituted into the fixed tap-length diffusion maximum correntropy criterion algorithm, and thus a diffusion variable tap-length maximum correntropy criterion (DVTMCC) algorithm is proposed. By communicating with the nodes in the neighborhood, the proposed algorithm realizes the information fusion of the entire network by means of diffusion, which has advantages of a high estimation accuracy, a small calculation cost, etc. Simulation experiments compare the convergence performance of DVTMCC algorithm and other distributed variable tap-length adaptive filtering algorithms, and fix tap-length diffusion maximum correntropy criterion algorithm under the impulsive noise environment. Simulation results show that the DVTMCC algorithm can estimate the tap-length and weight vector of the unknown vector at the same time under the impulsive noise environment, and its performance is better than compared algorithms.

**Keywords** Adaptive networks, Diffusion strategy, Maximum correntropy criterion, Variable tap-length, Impulsive noises

### 1 引言

分布式自适应滤波器利用分布式网络,通过空间不同位

置上的多个节点收集到的信息来共同估计未知向量,其工作原理与单节点的自适应滤波器有很大不同。与单节点的自适应滤波器相比,分布式网络中的节点与邻域内的节点协作进

行参数估计,因此分布式网络具有更快的收敛速度,被广泛应用于无线传感器网络、分布式在线模式识别等数据呈空间分布的应用领域<sup>[1-3]</sup>。分布式网络是由分布在空间不同位置上的节点组成的通信网络,网络内各节点通过特定的规则相互连接,其中,自适应网络是节点具备自适应处理能力的分布式网络。分布在空间上不同位置的节点之间主要采用3种协作策略,分别是:增量式策略<sup>[4-5]</sup>、共识式策略<sup>[6]</sup>以及扩散式策略<sup>[7-8]</sup>。在增量式自适应网络中,节点的信息会从一个节点传递到相邻的下一个节点,因此节点之间需要构成哈密顿环路。虽然增量式策略所需要的通信量很少,但是由于信息是按序传递的,当增量式网络中的某一节点发生故障时,整个环路都将瘫痪。经典的增量式自适应滤波算法有增量式最小均方(Incremental Least Mean Square, ILMS)算法<sup>[4]</sup>和增量式递归最小二乘(Incremental Recursive Least Square, IRLS)算法<sup>[5]</sup>。在共识式自适应网络和扩散式自适应网络中,与一个节点直接相连的所有节点的集合(包括该节点本身)被称为该节点的邻域,每个节点与其邻域内的其他节点交换信息。相比递增式策略,虽然共识式策略和扩散式策略需要更多的通信量,但是节点通过和邻域内的其他节点交换信息能够获得更高的估计精度和更快的速度,同时整个网络将更加稳健。值得注意的是,采用扩散式策略的自适应网络比采用共识式策略的自适应网络具有更加宽松的稳定性条件,因此扩散式策略是目前分布式网络中被研究最多、应用最广的协作策略。经典的扩散式自适应滤波算法包括扩散式最小均方(Diffusion Least Mean Square, DLMS)算法<sup>[7]</sup>和扩散式递归最小二乘(Diffusion Recursive Least Square, DRLS)算法<sup>[8]</sup>。

目前大多数分布式自适应滤波算法<sup>[4-8]</sup>只有在未知向量的阶数已知且不变的情况下才能达到相应的估计精度和速度,这类分布式自适应滤波算法被称为固定阶数的分布式自适应滤波算法。分布式自适应滤波算法的阶数设置会直接影响算法的收敛性能,如果设置的滤波器阶数小于未知向量的阶数,那么对未知向量的估计会产生由于建模不充分而导致的额外均方误差(Excess Mean Square Error, EMSE);如果设置的滤波器阶数过大,则会导致计算量过大,从而导致收敛速度变慢,那么对未知向量的估计会产生由自适应噪声而导致的EMSE<sup>[9]</sup>。因此,固定阶数的分布式自适应滤波算法不适用于阶数未知或者时变的情况。

在实际应用中,未知向量的阶数往往是未知或时变的,例如针对多用户回声消除系统和无线通信信道的辨识,随着发送装置和接收位置的改变,待辨识系统的权值和信道的阶数和权值都将发生改变<sup>[10-12]</sup>。分布式变阶数自适应滤波算法是解决上述问题的有效途径。文献[13]提出了一种扩散式变阶数最小均方(Diffusion Variable Tap-length, DVT)算法,该算法在对未知向量的阶数和权值进行估计的过程中使用了分数阶数,其由于无须预先根据未知向量的先验知识设定滤波器的阶数,具有操作更灵活、适用性更广等优点。文献[14]提出了一种空间分布式变阶数归一化最小均方算法(Spatially Distributed Variable Tap-length Normalized Least Mean Square, SDVTNLMS)算法,其根据最小化均方误差准则定义

了网络中各节点处理理论上的最优滤波器阶数,并基于扩散信息融合的思想,推导出了SDVTNLMS算法。该算法能够解决在阶数未知或时变条件下的未知向量估计问题。仿真结果表明,在高信噪比和低信噪比条件下,SDVTNLMS算法都能有效估计和跟踪未知参数向量的阶数和权值,且与固定阶数的自适应滤波算法相比,具有更好的收敛性能。然而上述算法<sup>[13-14]</sup>都是根据最小化MSE准则推导得到的,在非高斯噪声环境下这些算法的收敛性能会变差,甚至发散。实际环境中存在许多非高斯噪声(如脉冲噪声),上述算法的收敛性能会受到很大影响,因此本文认为有必要提出能在脉冲噪声环境下稳定收敛的分布式变阶数自适应滤波算法。

针对上述问题,本文提出了一种基于最大相关熵准则的分布式变阶数自适应滤波算法,称为扩散式变阶数最大相关熵准则算法。DVTMCC算法利用最大相关熵准则的强鲁棒性和低计算复杂度对未知向量的阶数和权值同时进行迭代更新,提高了在脉冲噪声下分布式变阶数自适应滤波算法的收敛性能。

## 2 自适应网络滤波基础

### 2.1 自适应网络

扩散式自适应网络如图1所示。

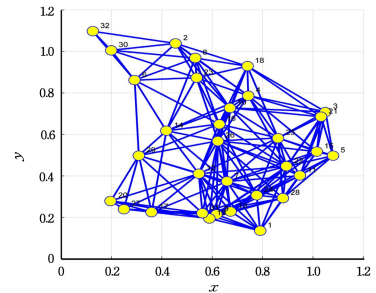


图1 扩散式自适应网络

Fig. 1 Diffusion adaptive network

在扩散式自适应网络中,与每个节点 $k$ 直接相连的所有节点的集合(包括节点 $k$ 本身)称为节点 $k$ 的邻域,节点 $k$ 可与邻域 $N_k$ 内的节点进行信息的双向传递,因此分布式自适应滤波算法比单节点的自适应滤波算法具有更好的收敛性能。自适应网络的目的是利用所有节点处的信息联合估计未知向量,假设自适应网络中输入信号和期望信号满足如下线性模型:

$$d_k(i) = \mathbf{u}_{k,M}^T(i) \mathbf{w}_{opt,M} + v_k(i) \quad (1)$$

其中, $d_k(i)$ 为节点 $k$ 处 $i$ 时刻的期望信号; $\mathbf{u}_{k,M}(i)$ 为节点 $k$ 处 $i$ 时刻的输入信号; $(\cdot)^T$ 为向量或矩阵的转置; $M$ 为待估计参数向量 $\mathbf{w}_{opt,M}$ 的阶数; $v_k(i)$ 为量测噪声,取自节点 $k$ 处 $i$ 时刻空间上相互独立的零均值、独立同分布的白噪声序列,其序列方差为 $\sigma_{v_k}^2$ 。

实际环境中未知向量的阶数往往是未知或时变的,即阶数是变化的,因此需要对未知向量的阶数和权值同时进行估计。节点 $k$ 处 $i$ 时刻的阶数记为 $M_{k,i}$ ,节点 $k$ 处 $i$ 时刻的输出误差可表示为:

$$e_{k,i}(\mathbf{w}, M_{k,i}) = d_k(i) - \mathbf{u}_{k,M_{k,i}}^T(i) \mathbf{w}_{M_{k,i}} \quad (2)$$

其中,  $\mathbf{w}_{M_{k,i}}$  为滤波器权值向量。

## 2.2 基于扩散 MSE 的最优阶数定义

文献[14]通过最小化 MSE 准则,使网络中各节点的 MSE 之和最小,以自适应更新未知向量的权值和阶数。假设所有的信号均为实值信号,节点  $k$  处的输入序列  $\mathbf{u}_k(i)$  为空间上相互独立的零均值、独立同分布的白噪声序列,序列方差为  $\sigma_{u,k}^2$ 。定义  $i$  时刻的全局 MSE 为  $i$  时刻网络中所有节点的 MSE 之和,即:

$$\mathbf{MSE}_i^{glob}(\mathbf{w}, M) = \sum_{k=1}^N \mathbb{E}[e_{k,i}^2(\mathbf{w}, M)] \quad (3)$$

其中,  $\mathbf{MSE}_i^{glob}$  表示  $i$  时刻的全局 MSE,  $\sum(\cdot)$  表示求和,  $N$  为网络中的总节点数,  $\mathbb{E}[\cdot]$  表示期望。由式(3)可知,  $\mathbf{MSE}_i^{glob}$  汇聚了网络中所有节点的信息进行集中处理,计算量大且可操作性差,因此不适合实际应用。根据自适应网络结构,定义节点  $k$  处  $i$  时刻的局部 MSE 为节点  $k$  的邻域  $N_k$  内所有节点的 MSE 的加权和,即:

$$\mathbf{MSE}_{k,i}^{loc}(\mathbf{w}, M) = \sum_{l \in N_k} c_{l,k} \mathbb{E}[e_{l,i}^2(\mathbf{w}, M)] \quad (4)$$

其中,  $c_{l,k} (\forall l, k=1, 2, \dots, N)$  为联合系数,且满足:

$$c_{l,k} = 0, l \in N_k, \mathbf{C}\mathbf{1} = \mathbf{1}, \mathbf{1}^T \mathbf{C} = \mathbf{1}^T \quad (5)$$

其中,  $\mathbf{1}$  为  $N \times 1$  维向量,向量中各元素均为 1;  $\mathbf{C}$  为联合矩阵,组成元素为  $c_{l,k}$ 。联合式(3)和式(4)可得,在  $i$  时刻的全局 MSE 和各节点处的局部 MSE 满足关系:

$$\mathbf{MSE}_i^{glob}(\mathbf{w}, M) = \sum_{k=1}^N \mathbf{MSE}_{k,i}^{loc}(\mathbf{w}, M) \quad (6)$$

相比全局 MSE,局部 MSE 的优点在于其各节点仅利用了其邻域内节点收集的信息,且其数据处理在本地进行,因此计算量小且可操作性强;局部 MSE 的缺点在于,网络中每个节点的信息只在相邻节点处得到了共享,其没有真正利用到全局的信息,因此无法保证估计精度。

由文献[14]可得节点  $k$  处的最优本地估计量为:

$$\mathbf{w}_{k,M}^{loc} = \left( \sum_{l \in N_k} c_{l,k} \mathbf{R}_{l,u} \right)^{-1} \left( \sum_{l \in N_k} c_{l,k} \mathbf{r}_{l,du} \right) \quad (7)$$

假设  $\mathbf{w}_{k,M}^{loc} \approx \mathbf{w}_{opt,M}$  成立,由式(7)可以定义另一种形式的扩散 MSE:

$$\mathbf{MSE}_{k,i}^{loc2}(\mathbf{w}, M) = \|\mathbf{w}_{k,M} - \mathbf{w}_{k,i}^{loc}\|_{\Sigma}^2 + \text{mmse} \quad (8)$$

其中,  $\text{mmse}$  是与权值无关的常数项;  $\|\cdot\|_{\Sigma}^2$  表示加权矢量范数 (Weighted Vector Norm, WVN), 对任意的自共轭矩阵 (Hermitian)  $\Sigma > 0$ , 满足  $\|a\|_{\Sigma}^2 = a^T \Sigma a$ 。

式(4)利用了相邻节点处的输入与期望信号,式(8)利用了各节点处的本地估计量  $\mathbf{w}_{k,M}^{loc}$ , 而因为  $\mathbf{w}_{k,M}^{loc}$  又利用了相邻节点处的输入与期望信号,所以式(4)和式(8)是等价的。因此可定义网络中节点  $k$  处  $i$  时刻的扩散 MSE 为:

$$\mathbf{MSE}_{k,i}^{dist}(\mathbf{w}, M) = \sum_{l \in N_k} c_{l,k} \mathbb{E}[e_{l,i}^2(\mathbf{w}, M)] + \sum_{l \in N_k/(k)} b_{l,k} \|\mathbf{w}_M - \mathbf{w}_{l,M}^{loc}\|_{\Sigma}^2 \quad (9)$$

其中,  $\|\cdot\|_{\Sigma}^2$  表示向量的欧几里得范数,  $b_{l,k} (\forall l, k=1, 2, \dots, N)$  为联合系数,且满足:

$$b_{l,k} = 0, l \notin N_k, \mathbf{1}^T \mathbf{B} = \mathbf{1}^T \quad (10)$$

其中,  $\mathbf{B}$  为联合系数矩阵。

节点  $k$  处  $i$  时刻的扩散 MSE 为相邻区域内所有节点处的局部 MSE 之和。扩散 MSE 的定义式表明,扩散 MSE 利

用了邻域内的输入与期望信号,而邻域各节点处的本地估计量又分别利用了各自邻域内节点的输入与期望信号,因此自适应网络中各节点的信息通过式(9)扩散到节点  $k$  处,即整个自适应网络的信息在节点  $k$  处得到了共享。在本地节点处理整个网络的数据信息,既减小了计算量,增加了可操作性,又提高了估计精度。因此,定义节点  $k$  处的最优滤波器阶数的代价函数为:

$$M_k^{opt} = \min\{M | \mathbf{MSE}_{k,\infty}^{dist}(\mathbf{w}, M - \Delta | M) - \mathbf{MSE}_{k,\infty}^{dist}(\mathbf{w}, M) \leq \zeta, M \geq \Delta\} \quad (11)$$

其中,  $\min(\cdot)$  为求最小值;  $\mathbf{MSE}_{k,\infty}^{dist}$  为稳态 ( $i \rightarrow \infty$ ) 扩散 MSE;  $M - \Delta | M$  表示  $M \times 1$  维向量的前  $M - \Delta$  个元素;  $\Delta$  为正整数,用于避免算法收敛于次优阶数<sup>[15]</sup>; 参数  $\zeta > 0$  由估计精度决定。

## 3 扩散式变阶数最大相关熵准则算法

本节以最大相关熵准则为代价函数,通过最大化各节点处的相关熵 (Correntropy) 之和,重新推导出基于最大相关熵准则的最优阶数定义,随后根据固定阶数的扩散式最大相关熵准则 (Diffusion Maximum Correntropy Criterion, DMCC) 算法<sup>[16]</sup>,提出了一种扩散式变阶数最大相关熵准则算法。

### 3.1 基于扩散 MCC 的最优阶数定义

相关熵是两个随机变量之间的广义相似性,令  $e_k(i) = x - y$ , 则相关熵可定义为:

$$V[e_k(i)] = \mathbb{E}[\kappa_{\sigma}[e_k(i)]] \quad (12)$$

其中,  $\mathbb{E}[\cdot]$  表示取期望;  $\kappa_{\sigma}[\cdot]$  表示核函数,最常用的核函数为高斯核函数。

不同于文献[14]中定义的基于扩散 MSE 的最优阶数定义,本文利用最大相关熵准则 (Maximum Correntropy Criterion, MCC), 定义  $i$  时刻的全局 MCC 为  $i$  时刻自适应网络中所有节点处的相关熵之和:

$$\mathbf{MCC}_i^{glob}(\mathbf{w}, M) = \sum_{k=1}^N \mathbb{E}[\kappa_{\sigma}[e_k(i)]] \quad (13)$$

接着定义  $i$  时刻节点  $k$  处的局部 MCC 为其邻域  $N_k$  内所有节点处的 MCC 的加权和:

$$\mathbf{MCC}_{k,i}^{loc}(\mathbf{w}, M) = \sum_{l \in N_k} c_{l,k} \mathbb{E}[\kappa_{\sigma}[e_k(i)]] \quad (14)$$

其中,  $\mathbf{MCC}_{k,i}^{loc}$  为节点  $k$  处  $i$  时刻的局部 MCC。本文根据 Metropolis 规则计算  $c_{l,k}$ , 即:

$$c_{l,k} = \begin{cases} \frac{1}{\max(n_k, n_l)}, & l \in N_k \text{ and } l \neq k \\ 1 - \sum_{l \in N_k/(k)} c_{l,k}, & l = k \\ 0, & l \notin N_k \end{cases} \quad (15)$$

其中,  $n_k = \deg(N_k)$  表示节点  $k$  的邻域  $N_k$  中的节点数,  $N_k / \{k\}$  表示节点  $k$  的邻域  $N_k$  中除了节点  $k$  之外的所有节点。全局 MCC 和局部 MCC 满足:

$$\mathbf{MCC}_i^{glob} = \sum_{k=1}^N \mathbf{MCC}_{k,i}^{loc}(\mathbf{w}, M) \quad (16)$$

同理,  $\mathbf{MCC}_{k,i}^{loc}$  意味着只有邻域内的节点信息得到了共享,节点  $k$  无法充分利用网络中其他节点处的信息。式(9)综合利用节点处的输入、期望信号和各节点处的本地估计量,又通过本地估计量融合邻域内其他节点的信息,进而将节点  $k$  的

邻域中其他节点的本地估计量汇聚到节点  $k$  处,即式(9)利用了整个网络的数据信息。因此,定义扩散 MCC 如下:

$$\mathbf{MCC}_{k,i}^{dist}(\mathbf{w}, M) = \sum_{l \in N_k} c_{l,k} \mathbb{E}[\kappa_\sigma[e_k(i)]] + \sum_{l \in N_k / \{k\}} b_{l,k} \|\mathbf{w}_M - \mathbf{w}_{l,M}^{loc}\|_2^2 \quad (17)$$

其中,  $b_{l,k}$  为联合系数。通常情况下根据相关度原则计算  $b_{l,k}$ , 即:

$$b_{l,k} = \begin{cases} \frac{n_l}{\sum_{i \in N_k} n_i}, & l \in N_k \\ 0, & \text{other} \end{cases} \quad (18)$$

综上所述,基于扩散 MCC 的定义,可得到节点  $k$  处的最优滤波器阶数的定义为:

$$M_k^{opt2} = \min\{M | \mathbf{MCC}_{k,\infty}^{dist}(\mathbf{w}, M - \Delta) - \mathbf{MCC}_{k,\infty}^{dist}(\mathbf{w}, M)\} \quad (19)$$

其中,  $M_k^{opt2}$  为节点  $k$  处的最优滤波器阶数,  $\mathbf{MCC}_{k,\infty}^{dist}$  为稳态 ( $i \rightarrow \infty$ ) 扩散 MCC,  $M - \Delta | M$  表示  $M \times 1$  维向量的前  $M - \Delta$  个元素。

### 3.2 DVTMCC 算法

为了避免分布式变阶数自适应滤波算法出现收敛缓慢的问题,在阶数  $M$  的迭代方程中引入泄露因子  $\alpha_k$  ( $0 < \alpha_k \ll 1$ )<sup>[17-18]</sup>。但是泄露因子  $\alpha_k$  可能导致收敛过程中滤波器阶数  $M$  不是整数,因此需要引入分数阶数来进行阶数的实时迭代,当分数阶数的变化超出限定的范围后,滤波器阶数的值会被设定为相应的整数。由文献[16-18]可得节点  $k$  在  $i$  时刻的阶数迭代方程为:

$$M_{k,f}(i+1) = M_{k,f}(i) - \alpha_k - \gamma_k \{\mathbf{MCC}_{k,i}^{dist}(\mathbf{w}_k(i), M_{k,i}) - \mathbf{MCC}_{k,i}^{dist}(\mathbf{w}_k(i), M_{k,i} - \Delta)\} \quad (20)$$

$$M_{k,i+1} = \begin{cases} \lfloor M_{k,f}(i+1) \rfloor, & |M_{k,f}(i+1) - M_{k,i}| \geq \delta_k \\ M_{k,i}, & \text{other} \end{cases} \quad (21)$$

其中,  $M_{k,f}(i+1)$  为节点  $k$  处当前时刻估计的分数阶数; 正整数  $M_{k,i+1}$  为节点  $k$  当前时刻的真实阶数;  $\gamma_k$  为滤波器阶数的收敛步长, 满足  $\alpha_k \ll \gamma_k$ ;  $\lfloor \cdot \rfloor$  表示向下取整;  $|\cdot|$  表示绝对值;  $\delta_k$  是一个较小的整数阈值, 用于防止滤波器阶数频繁变化。

由式(17)可知, 扩散 MCC 的定义分为以下两部分:

$$\sum_{l \in N_k} c_{l,k} \mathbb{E}[\kappa_\sigma[e_k(i)]] \quad (22)$$

$$\sum_{l \in N_k / \{k\}} b_{l,k} \|\mathbf{w}_M - \mathbf{w}_{l,M}^{loc}\|_2^2 \quad (23)$$

式(22)为自适应融合, 式(23)为空间融合。扩散式自适应滤波算法按照先自适应融合后空间融合的顺序 (Adapt-then-combine, ATC) 进行权值迭代的算法, 被称为 ATC 式自适应滤波算法; 而按照先空间融合后自适应融合的顺序进行权值迭代的算法, 被称为 CTA 式自适应算法。文献[19]已经证实了 ATC 式自适应滤波算法比 CTA 式自适应滤波算法融合了更多的信息, 具有稍好的收敛性能。同理, 阶数的迭代过程也分为 ATC 式和 CTA 式, 本文采用 ATC 式的阶数迭代公式, 即:

$$L_{k,f}(i+1) = M_{k,f}(i) - \alpha_k - \gamma_k \sum_{l \in N_k} c_{l,k} \times \{\kappa_\sigma[e_{l,i}(\mathbf{w}_k(i), M_{k,i})] - \kappa_\sigma[e_{l,i}(\mathbf{w}_k(i), M_{k,i} - \Delta)]\} \quad (24)$$

$$M_{k,f}(i+1) = L_{k,f}(i+1) - \gamma_k \sum_{l \in N_k / \{k\}} b_{l,k} [\|\mathbf{w}_{k,M_{k,i}}(i) - \mathbf{w}_{l,M_{k,i}}^{loc}\|_2^2 - \|\mathbf{w}_{k,M_{k,i}-\Delta} - \mathbf{w}_{l,M_{k,i}-\Delta}^{loc}\|_2^2] \quad (25)$$

$$L_{k,i+1} = \begin{cases} \lfloor L_{k,f}(i+1) \rfloor, & |L_{k,f}(i+1) - L_{k,i}| \geq \delta_k \\ L_{k,i}, & \text{other} \end{cases} \quad (26)$$

其中,  $M_{k,f}(i+1)$  为节点  $k$  当前时刻估计的分数阶数,  $L_{k,f}(i+1)$  为节点  $k$  当前时刻的本地分数阶数,  $L_{k,i+1}$  为节点  $k$  当前时刻的本地整数阶数。采用瞬时值来近似期望值, 并由欧几里得范数的性质, 将式(25)改写为:

$$M_{k,f}(i+1) = L_{k,f}(i+1) - \gamma_k \sum_{l \in N_k / \{k\}} b_{l,k} \|\mathbf{w}_{k,\Delta|M_{k,i}}(i+1) - \mathbf{w}_{l,\Delta|M_{k,i}}^{loc}\|_2^2 \quad (27)$$

$l_2$  范数不适用于非高斯噪声环境, 而  $l_1$  范数具有对非高斯噪声的强鲁棒性, 因此将式(27)中的  $l_2$  范数替换为  $l_1$  范数, 即:

$$M_{k,f}(i+1) = L_{k,f}(i+1) - \gamma_k \sum_{l \in N_k / \{k\}} b_{l,k} \|\mathbf{w}_{k,\Delta|M_{k,i}}(i+1) - \mathbf{w}_{l,\Delta|M_{k,i}}^{loc}\|_1 \quad (28)$$

此时, 由固定阶数的 DMCC 算法可得到节点  $k$  处的本地权向量估计为:

$$\boldsymbol{\varphi}_{l,M_{k,i}}(i+1) = \mathbf{w}_{l,M_{k,i}}(i) + \mu_l \sum_{s \in N_l} c_{s,l} \kappa_\sigma[d_s(i) - \mathbf{u}_{s,M_{k,i}}^T(i) \mathbf{w}_{l,M_{k,i}}(i)] \times \mathbf{u}_{s,M_{k,i}}(i) [d_s(i) - \mathbf{u}_{s,M_{k,i}}^T(i) \mathbf{w}_{l,M_{k,i}}(i)] \quad (29)$$

其中, 估计量  $\boldsymbol{\varphi}_{l,M_{k,i}}(i+1)$  为节点  $l$  当前时刻的本地权值向量估计。由式(29)可知, 因为节点  $\boldsymbol{\varphi}_{l,M_{k,i}}(i+1)$  融合了当前时刻节点  $l$  邻域内各节点的输入和期望信号, 所以其是比本地估计量  $\mathbf{w}_{l,M_{k,i}}^{loc}$  更好的选择, 因此用  $\boldsymbol{\varphi}_{l,\Delta|M_{k,i}}(i+1)$  替换  $\mathbf{w}_{l,\Delta|M_{k,i}}^{loc}$ 。同理, 可将  $\mathbf{w}_{k,\Delta|M_{k,i}}(i)$  替换为  $\boldsymbol{\varphi}_{k,\Delta|M_{k,i}}(i+1)$ 。同样的, 节点  $k$  当前时刻的本地阶数估计  $L_{k,i+1}$  利用了网络中当前时刻的信息, 是一个比  $M_{k,i}$  更优的阶数估计量, 因此将  $M_{k,i}$  替换为  $L_{k,i+1}$ 。综上所述, 可将式(28)改写为:

$$M_{k,f}(i+1) = L_{k,f}(i+1) - \gamma_k \sum_{l \in N_k / \{k\}} b_{l,k} \|\boldsymbol{\varphi}_{k,\Delta|L_{k,i+1}}(i+1) - \boldsymbol{\varphi}_{l,\Delta|L_{k,i+1}}\|_1 \quad (30)$$

由式(21)可得到节点  $k$  当前时刻的滤波器阶数  $M_{k,i+1}$ 。最后, 以  $M_{k,i+1}$  作为滤波器阶数, 根据固定阶数的 DMCC 算法, 得到节点  $k$  当前时刻的滤波器权向量迭代公式如下:

$$\begin{cases} \boldsymbol{\varphi}_{k,M_{k,i+1}}(i+1) = \mathbf{w}_{k,M_{k,i+1}}(i) + \mu_k \sum_{l \in N_k} c_{l,k} \kappa_\sigma[d_l(i) - \mathbf{u}_{l,M_{k,i+1}}^T(i) \mathbf{w}_{k,M_{k,i+1}}(i)] \times \mathbf{u}_{l,M_{k,i+1}}(i) [d_l(i) - \mathbf{u}_{l,M_{k,i+1}}^T(i) \mathbf{w}_{k,M_{k,i+1}}(i)] \\ \boldsymbol{\varphi}_{l,M_{k,i+1}}(i+1) = \sum_{i \in N_k} b_{l,k} \boldsymbol{\varphi}_{l,M_{k,i+1}}(i+1) \end{cases} \quad (31)$$

最后, 本文使用的核函数是文献[20]中提到的改进高斯核函数, 是比经典高斯核函数更好的选择, 其定义式为:

$$\kappa_\sigma(e_k(i)) = \frac{e_k^2(i)}{\sigma^2} \exp\left(-\frac{e_k^2(i)}{\sigma^2}\right) \quad (32)$$

其中,  $\sigma$  表示核宽。

现总结本文所提的 DVTMCC 算法如下: 首先, 根据式(15)和式(18)确定联合矩阵  $\mathbf{C}$  及联合矩阵  $\mathbf{B}$ 。其次, 确定迭代初值  $\{M_{k,f}(0) = M_{k,0} > \Delta, \mathbf{w}_{k,M_{k,0}} = \mathbf{0}\}$ , 其中  $\mathbf{0}$  为与  $\mathbf{W}$  相同维度的零向量。在每个迭代时刻  $i$  循环执行以下

步骤,直至算法达到稳态。

步骤1 计算相邻节点 $\{l \in N_k\}$ 的输出误差。

$$e_{l,i}(\mathbf{w}_k(i), M_{k,i}) = d_l(i) - \mathbf{u}_{l,M_{k,i}}^T(i) \mathbf{w}_{k,M_{k,i}}(i) e_{l,i}(\mathbf{w}_k(i), M_{k,i} - \Delta) = d_l(i) - \mathbf{u}_{l,M_{k,i} - \Delta}^T(i) \mathbf{w}_{k,M_{k,i} - \Delta}(i) \quad (33)$$

步骤2 滤波器权值向量阶数的自适应融合。

$$L_{k,f}(i+1) = M_{k,f}(i) - \alpha_k - \gamma_k \sum_{l \in N_k} c_{l,k} \times \{\kappa_\sigma [e_{l,i}(\mathbf{w}_k(i), M_{k,i})] - \kappa_\sigma [e_{l,i}(\mathbf{w}_k(i), M_{k,i} - \Delta)]\} \\ L_{k,i+1} = \begin{cases} |L_{k,f}(i+1)|, & |L_{k,f}(i+1) - L_{k,i}| \geq \delta_k \\ L_{k,i}, & \text{other} \end{cases} \quad (34)$$

步骤3 滤波器权向量权值的自适应融合。

$$\boldsymbol{\varphi}_{k,L_{k,i+1}}(i+1) = \mathbf{w}_{k,L_{k,i+1}}(i) + \mu_k \sum_{l \in N_k} \kappa_\sigma [d_l(i) - \mathbf{u}_{l,L_{k,i+1}}^T(i) \mathbf{w}_{k,L_{k,i+1}}(i)] \times \mathbf{u}_{l,L_{k,i+1}}(i) [d_l(i) - \mathbf{u}_{l,L_{k,i+1}}^T(i) \mathbf{w}_{k,L_{k,i+1}}(i)] \quad (35)$$

步骤4 滤波器权值向量阶数的空间融合。

$$M_{k,f}(i+1) = L_{k,f}(i+1) - \gamma_k \sum_{l \in N_k / \{k\}} b_{l,k} \|\boldsymbol{\varphi}_{k,\Delta|L_{k,i+1}}(i+1) - \boldsymbol{\varphi}_{l,\Delta|L_{k,i+1}}\|_1^2 \\ M_{k,i+1} = \begin{cases} \lfloor M_{k,f}(i+1) \rfloor, & |M_{k,f}(i+1) - L_{k,i+1}| \geq \delta_k \\ L_{k,i+1}, & \text{other} \end{cases} \quad (36)$$

步骤5 滤波器权向量的迭代。

$$\begin{cases} \boldsymbol{\varphi}_{k,M_{k,i+1}}(i+1) = \mathbf{w}_{k,M_{k,i+1}}(i) + \mu_k \sum_{l \in N_k} \kappa_\sigma [d_l(i) - \mathbf{u}_{l,M_{k,i+1}}^T(i) \mathbf{w}_{l,M_{k,i+1}}(i)] \times \mathbf{u}_{l,M_{k,i+1}}(i) [d_l(i) - \mathbf{u}_{l,M_{k,i+1}}^T(i) \mathbf{w}_{l,M_{k,i+1}}(i)] \\ \mathbf{w}_{k,M_{k,i+1}}(i+1) = \sum_{l \in N_k} b_{l,k} \boldsymbol{\varphi}_{l,M_{k,i+1}}(i+1) \end{cases} \quad (37)$$

## 4 仿真实验与分析

为了测试所提的 DVTMCC 算法的收敛性能,本节设计了两个仿真实验并测试了算法在脉冲噪声下的收敛性能。

首先,如图1所示构建自适应网络。输入信号 $\mathbf{u}_k$ 和量测噪声信号 $\mathbf{u}_k$ 由空间上零均值、独立同分布的白噪声序列通过AR进程 $G(z) = (1 + 0.6z^{-1}) / (1 + z^{-1} + 0.21z^{-2})$ 产生。网络中各节点处的输入序列方差 $\sigma_{u_{l,k}}^2$ 如图2所示。仿真过程中添加脉冲噪声 $\mathbf{I}_k$ ,其模型为伯努利高斯噪声模型,方差为 $\sigma_{I_{l,k}}^2 = 1000\sigma_{u_{l,k}}^2$ 。

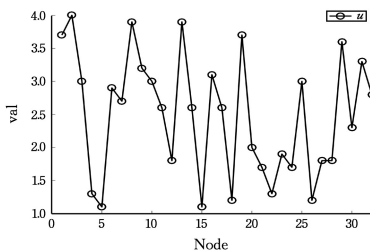


图2 各节点处的输入序列方差

Fig. 2 Input sequence variance at each node

斯白噪声信号 $\mathbf{v}_k$ 的方差比值,其定义式为:

$$\text{SNR}_k = 10 \lg \left( \frac{\text{var}(\mathbf{u}_k^T \mathbf{u})}{\sigma_{v,k}^2} \right) \quad (38)$$

仿真过程中用均方偏差(Mean Square Deviation, MSD)评估算法的收敛性能,其定义式为:

$$\text{MSD}_k(i+1) = 10 \lg \left( \frac{1}{M} \sum_{j=1}^M [\rho \|\mathbf{w}_{opt} - \mathbf{w}_{k,j}(i)\|^2 + (1-\rho) \|\mathbf{w}_{opt} - \mathbf{w}_{k,j}(i+1)\|^2] \right) \quad (39)$$

其中, $\text{MSD}_k(i+1)$ 为节点 $k$ 当前时刻的均方偏差,平滑因子 $\rho = 0.99$ ,蒙特卡罗仿真次数 $Tri = 100$ ,实验中设置 $\|\mathbf{w}_{opt}\|_2^2 = 1$ ,因此MSD与归一化均方偏差(Normalized MSD, NMSD)等同。实验中采用补零操作以保持作差的权向量维数一致。

### 4.1 在脉冲噪声下 SDVTNLMS 算法、DVTMCC 算法和 DVTLMS 算法的比较

仿真实验一测试了 DVTLMS 算法、SDVTNLMS 算法和 DVTMCC 算法在脉冲噪声环境下的收敛性能和对阶数的跟踪估计能力。仿真环境设置在 $Pr = 0.01$ (发生概率)的脉冲噪声环境下,且在 $i = 7500$ 时,滤波器阶数由 $M = 60$ 突变为 $M = 120$ 。仿真参数设置如下: DVTMCC 算法的迭代初值为 $N_{\min} = N_{k,f}(0) = N_{k,0} = 31$ , $\mathbf{w}_{k,N_{k,0}} = \mathbf{0}$ ,权值的迭代步长 $\mu_k = 0.04$ ,滤波器阶数的迭代步长 $\gamma = 10$ ,参数 $\Delta = 20$ ,核宽 $\sigma = 0.5$ ,泄漏因子 $\alpha = 0.008$ ,整数阈值 $\delta = 2$ ,正则化因子 $\epsilon = 0.0001$ 。SDVTNLMS 算法的权值迭代步长 $\mu_k = 0.2$ ,滤波器阶数的迭代步长 $\gamma = 10$ ,参数 $\Delta = 20$ ,其余参数设置与 DVTMCC 算法相同。DVTLMS 算法的权值迭代步长 $\mu_k = 0.004$ ,滤波器阶数的迭代步长 $\gamma = 8$ ,参数 $\Delta = 20$ ,其余参数设置和 DVTMCC 算法相同。

DVTLMS 算法和 SDVTNLMS 算法的参数分别依据文献[13]和文献[14]进行设置。考虑到实验环境的差异,与步长 $\mu_k$ 、参数 $\gamma$ 和参数 $\Delta$ 的选取依据为:使 DVTLMS 算法、SDVTNLMS 算法和 DVTMCC 算法的权值和阶数收敛达到相近的收敛速度,以便于分析对比。实验中选择的参数是为了让3种算法保持相近的权值初始收敛速度和滤波器阶数收敛速度,以便公平比较。

图3给出了 SDVTNLMS 算法、DVTMCC 算法和 DVTLMS 算法的收敛性能对比。

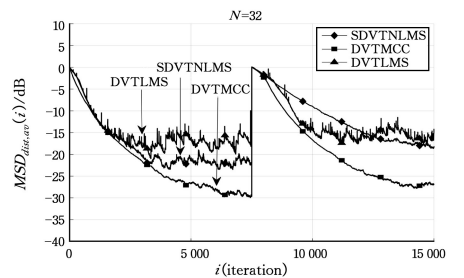


图3 SDVTNLMS、DVTMCC 和 DVTLMS 的收敛性能对比

Fig. 3 Comparison of convergence performance of SDVTNLMS, DVTMCC and DVTLMS

由图3可知,在脉冲噪声下,当未知向量的阶数发生变化时,DVTMCC 算法能有效地跟踪未知向量的变化,且在估计过程中保持较高的估计精度和较快的估计速度。而 SDVT-

NLMS算法和DVTLM算法在脉冲噪声环境下算法的收敛性能受到较大影响,在相同的收敛速度下,上述两种算法的稳态误差比DVTMCC算法大,并且在未知向量的阶数发生变化时,SDVTNLMS算法的收敛速度变慢。图4给出了收敛过程中阶数的变化,可以看出DVTMCC算法能较好地跟踪未知向量的阶数变化。

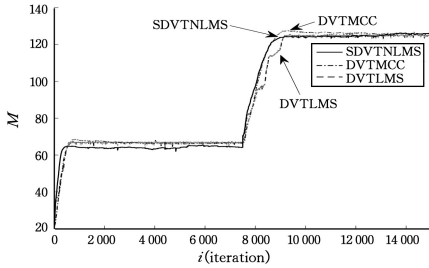


图4 仿真实验一阶数变化

Fig. 4 Change of tap-length in experiment one

#### 4.2 在脉冲噪声环境下DVTMCC算法和固定阶数的DMCC算法的比较

仿真实验二测试了DVTMCC算法和固定阶数的DMCC算法的跟踪性能和对阶数的跟踪估计能力,仿真实验在 $Pr=0.05$ 的脉冲噪声环境下进行,且在 $i=7500$ 时,滤波器阶数由 $M=60$ 突变为 $M=120$ 。DVTMCC算法的仿真参数设置与仿真实验一相同。阶数 $M=60$ 的DMCC算法和阶数 $M=120$ 的DMCC算法,权值的迭代步长都设置为 $\mu_k=0.2$ ,其余参数与DVTMCC算法相同。为了便于描述,阶数 $M=60$ 的DMCC算法记为DMCC1,阶数 $M=120$ 的DMCC算法记为DMCC2。DMCC算法的参数根据文献[16]进行设置,步长 $\mu_k$ 的选择依据是:为了公平比较,使所有的算法达到相近的稳态误差。

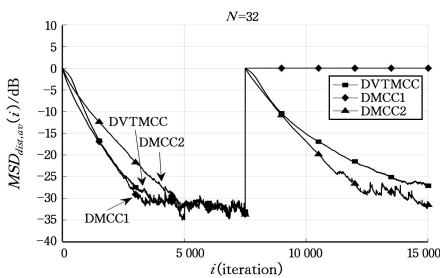


图5 DVTMCC算法与固定阶数的DMCC算法的收敛性能对比

Fig. 5 Convergence performance comparison between DVTMCC algorithm and fixed tap-length DMCC algorithms

由图5可知,在未知向量的阶数发生变化前,DMCC1,DMCC2和DVTMCC达到相近的稳态误差,其中,DMCC1具有最快的收敛速度,其次是DVTMCC算法,DMCC2最慢。此时最优阶数小于估计的阶数,这是因为自适应噪声导致DMCC2的收敛性能变差。在未知向量的阶数由 $M=60$ 变为 $M=120$ 后,DMCC1由于建模不足收敛性能受到极大影响,甚至不收敛。DMCC2在整个过程中都保持较高的估计精度,而DVTMCC算法对阶数的估计存在一定的误差,因此此时DMCC2的稳态误差稍好于DVTMCC算法。综合分析整

个过程,DVTMCC算法能够自适应跟踪未知向量阶数的变化,性能优于固定阶数的DMCC算法。图6给出了收敛过程中各算法对未知向量阶数的估计。

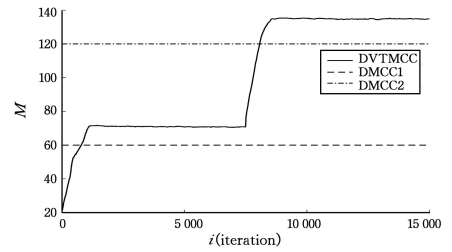


图6 仿真实验二的阶数变化

Fig. 6 Change of tap-length in experiment two

**结束语** 本文基于最大相关熵准则提出一种扩散式最大相关熵准则算法。该算法仅需要邻域内节点处的期望信号、输入信号与本地估计量,就能获取整个自适应网络中的信息,具有计算量小、估计精度高等优点。DVTMCC算法通过最大化相关熵准则,在脉冲噪声下能同时实现对未知向量阶数权值和阶数的估计。仿真实验结果表明,相比对比算法和固定阶数的DMCC算法,在脉冲噪声环境下DVTMCC算法能达到更低的稳态误差,提升了分布式变阶数自适应滤波算法的收敛性能。

本文研究仍然存在不足之处,例如代价函数是构建自适应滤波器的过程中非常重要的一环,选择的代价函数不同得到的权值向量的迭代公式也不同,根据不同的应用场景为自适应滤波器选择不同的代价函数是非常重要的。在高斯噪声环境下,MSE准则计算简单,是非常合适的选择;在脉冲噪声环境下,基于MSE准则的算法收敛性能恶化,而MCC在脉冲噪声环境下则表现出较强的鲁棒性,且计算复杂度低。除此之外还有许多不同的代价函数。本文将实验环境设置在脉冲噪声环境下,未来将继续研究在其他实验环境中如何选择合适的代价函数以进一步提高自适应滤波算法的收敛性能。

#### 参考文献

- [1] SAMPATHKUMAR A, MURUGAN S, ELNGAR A A, et al. A Novel Scheme for an IoT-Based Weather Monitoring System Using a Wireless Sensor Network[M]. Berlin: Springer, 2020: 181-191.
- [2] PANDA M, GOUDA B S. Distributed Online Fault Diagnosis in Wireless Sensor Networks[M]. Berlin: Springer, 2020: 197-221.
- [3] TAVAKOLIAN N, NAZEMI A, AZIMIFAR Z, et al. Face recognition under occlusion for user authentication and invigilation in remotely distributed online assessments [J]. International Journal of Intelligent Defence Support Systems, 2018, 5(4): 277-297.
- [4] KHALILI A, TINATI M A, RASTEGARNIA A. Steady-state analysis of incremental LMS adaptive networks with noisy links [J]. IEEE Transactions on Signal Processing, 2011, 59(5): 2416-2421.
- [5] KHALILI A, TINATI M A, RASTEGARNIA A. Analysis of incremental RLS adaptive networks with noisy links [J]. IEICE

- Electronics Express,2011,8(9):623-628.
- [6] TU S Y,SAYED A H. Diffusion strategies outperform consensus strategies for distributed estimation over adaptive networks [J]. IEEE Transactions on Signal Processing, 2012, 60 (12): 6217-6234.
- [7] CATTIVELLI F S,SAYED A H. Diffusion LMS strategies for distributed estimation[J]. IEEE Transactions on Signal Processing, 2009, 58(3): 1035-1048.
- [8] CATTIVELLI F S,LOPES C G,SAYED A H. Diffusion recursive least-squares for distributed estimation over adaptive networks [J]. IEEE Transactions on Signal Processing, 2008, 56(5): 1865-1877.
- [9] ZHANG Y,LI N,CHAMBERS J A, et al. Steady-state performance analysis of a variable tap-length LMS algorithm[J]. IEEE Transactions on Signal Processing, 2008, 56(2): 839-845.
- [10] ALBU I,ANGHEL C,PALEOLOGU C. Adaptive filtering in acoustic echo cancellation systems—A practical overview[C]// 2017 9th International Conference on Electronics, Computers and Artificial Intelligence (ECAD). IEEE, 2017: 1-6.
- [11] LI J,ZHANG L,FENG X, et al. Feature extraction and area identification of wireless channel in mobile communication[J]. Journal of Internet Technology, 2019, 20(2): 545-553.
- [12] AL-KINANI A,WANG C X,ZHOU L, et al. Optical wireless communication channel measurements and models [J]. IEEE Communications Surveys & Tutorials, 2018, 20(3): 1939-1962.
- [13] LI L,FENG J,HE J, et al. A distributed variable tap-length algorithm within diffusion adaptive networks[J]. AASRI Proceedings, 2013, 5(1): 77-84.
- [14] ZHANG Y G,WANG C C,WEI Y, et al. A spatially distributed variable tap-length strategy over adaptive networks[J]. Acta Automatica Sinica, 2014, 40(7): 1355-1365.
- [15] GONG Y,COWAN C F N. An LMS style variable tap-length algorithm for structure adaptation[J]. IEEE Transactions on Signal Processing, 2005, 53(7): 2400-2407.
- [16] MA W T,CHEN B D,DUAN J D, et al. Diffusion Maximum Correntropy Criterion Algorithms for Robust Distributed Estimation[J]. Digital Signal Processing, 2016, 58(1): 10-19.
- [17] HAYKIN S. 自适应滤波器原理(第四版)[M]. 郑宝玉,译. 北京:电子工业出版社, 2002: 179-369.
- [18] MAYYAS K,ABOULNASR T. Leaky LMS algorithm: MSE analysis for Gaussian data[J]. IEEE Transactions on Signal Processing, 1997, 45(4): 927-934.
- [19] TAKAHASHI N,YAMADA I,SAYED A H. Diffusion least-mean squares with adaptive combiners[C]// 2009 IEEE International Conference on Acoustics, Speech and Signal Processing. Taipei: IEEE Press, 2009(1): 2845-2848.
- [20] WANG W,ZHAO J,QU H, et al. A correntropy inspired variable step-size sign algorithm against impulsive noises[J]. Signal Processing, 2017, 141(1): 168-175.



**LIN Yun**, born in 1972, Ph.D, associate professor. His main research interests include adaptive filtering and sparse adaptive filtering.



**HUANG Zhen-hang**, born in 1997, post-graduate. His main research interests include adaptive filtering and distributed adaptive filtering.