

DOI:10.16356/j.1005-2615.2015.01.007

基于多级维纳滤波器的树型 WSN 分布式线性约束 最小方差波束形成方法

黄庆东 卢光跃 庞胜利 包志强

(西安邮电大学通信与信息工程学院,西安,710121)

摘要:为减少树型无线传感器网络(Wireless sensor network, WSN)中分布式线性约束最小方差(Linearly constrained minimum variance, LCMV)波束形成器的计算量,将多级分解技术用于 WSN 分布式波束形成技术中,提出基于多级维纳滤波器(Multistage Wiener filter, MSWF)的分布式 LCMV 波束形成器方法。该方法通过有效引入 MSWF 技术避免本地协方差矩阵估计及求逆运算,能以更少的计算量获得分布式 LCMV 波束形成器相同的输出性能,说明新方法继承了 MSWF 和分布式 LCMV 波束形成器的优点。计算机仿真结果验证了算法的优良性能。

关键词:无线传感器网络;LCMV 波束形成器;多级维纳滤波器;分布式信号估计

中图分类号: TN911.23 **文献标志码:** A **文章编号:** 1005-2615(2015)01-0052-07

Multistage Wiener Filter Based Distributed LCMV Beamforming Method in Tree Topology WSN

Huang Qingdong, Lu Guangyue, Pang Shengli, Bao Zhiqiang

(School of Communication and Information Engineering, Xi'an University of
Posts and Telecommunications, Xi'an, 710121, China)

Abstract: To reduce the calculation of distributed linearly constrained minimum variance (D-LCMV) beamforming in tree topology wireless sensor network (WSN), a multistage Wiener filter (MSWF) based D-LCMV beamforming method is proposed by introducing a multistage decomposition techniques. By effectively applying the MSWF technology to the new method, the estimation and the inversion of local covariance matrix are avoided. The new method can reach the same output performance as that of the D-LCMV beamformer with less amount of calculation. Therefore it is shown that the new method inherits advantages of both MSWF and D-LCMV beamformer. Simulation results prove that the new method achieves excellent performance.

Key words: wireless sensor network (WSN); linearly constrained minimum variance (LCMV) beamformer; multistage Wiener filter (MSWF); distributed signal estimation(DSE)

无线传感器网络(Wireless sensor network, WSN)通过互连节点相互协作完成一定的信号处理任务。集中式处理方法^[1]容易获取最优输出,但往往不能够充分体现 WSN 网络的性能和特点。

分布式处理方法^[2-16]具有通信带宽小、节点处理能力要求较低且算法可扩展性强等特点,引起广大学者的普遍关注。分布式参数估计(Distributed parameter estimation, DPE)^[2-8]方法通过网络节点

基金项目:国家自然科学基金(61301091,61271276)资助项目;陕西省教育厅课题(11JK0929)资助项目。

收稿日期:2014-05-18;**修订日期:**2014-06-23

通信作者:黄庆东,男,副教授,E-mail:huangqingdong@xupt.edu.cn。

交互协作迭代估计一个固定长度的参数向量。分布式信号估计 (Distributed signal estimation, DSE)^[9,10,12,15] 类似于块处理算法对成批数据进行处理,它对所收集的传感器数据进行基于波束形成技术的网内压缩形成标量(单通道)信号进行信息交互。因此 DSE 算法相比 DPE 算法需要的通信带宽更小。在 DSE 算法方面,最优化分布式波束形成器可以通过自适应的途径进行解决^[9,12,13,15]。文献[9]研究了无约束分布式节点特定的信号估计;文献[13]将问题推广为线性约束节点特定的信号估计。在此基础上,文献[15]提出并研究了分布式线性约束最小方差(Distributed linearly constrained minimum variance, D-LCMV)波束形成方法。与文献[9,13]不同,文献[15]中每个节点具有相同的线性约束,它只需要单个通道用于标量压缩/融合信号传输,且每个节点的波束形成器输出相同。D-LCMV 在不需要计算全局协方差矩阵情况下,每个节点仍可获得与集中型 LCMV 波束形成器相同的输出。

本文对文献[15]的算法进行改进,提出了在树型 WSN 应用环境下基于多级维纳滤波器(Multi-stage Wiener filter, MSWF)^[17-18] 的 D-LCMV(D-LCMSWF)波束形成方法。新算法在继承原有算法优点的基础上,融入 MSWF 算法低复杂度、快收敛特性,可以避免本地节点局部协方差矩阵估计及求逆运算,进而有效降低了节点的运算量。仿真表明新算法在较小的运算量条件下,每个节点仍然能获得与 D-LCMV 波束形成器相同的输出。

1 集中型 LCMV 波束形成器

假设无线传感器网络由节点集合 $\kappa = \{1, 2, \dots, K\}$ 构成。节点 k 的 M_k 个不同的传感器接收到 M_k 维平稳随机信号 \mathbf{y}_k 。定义 M 维信号 \mathbf{y} 由节点信号 \mathbf{y}_k 堆叠构成,其中 $M = \sum_{k \in \kappa} M_k$ 。

集中型 LCMV 波束形成器 $\hat{\mathbf{w}}$ 由如下最优问题定义^[14]

$$\hat{\mathbf{w}} = \operatorname{argmin} E\{|\mathbf{w}^H \mathbf{y}|^2\} \quad \text{s. t.} \quad \mathbf{C}^H \mathbf{w} = \mathbf{f} \quad (1)$$

式中: \mathbf{C} 为 $M \times Q$ 约束矩阵; \mathbf{f} 为 Q 维非零响应矢量; $E\{\cdot\}$ 为求数学期望;上标 H 表示共轭转置。代价函数记为 $J(\mathbf{w}) = E\{|\mathbf{w}^H \mathbf{y}|^2\} = \mathbf{w}^H \mathbf{R}_{yy} \mathbf{w}$, 是波束形成器输出 $d = \mathbf{w}^H \mathbf{y}$ 的方差。假设 \mathbf{R}_{yy} 和 \mathbf{C} 是满秩的,集中型 LCMV 波束形成器的解为

$$\hat{\mathbf{w}} = \mathbf{R}_{yy}^{-1} \mathbf{C} (\mathbf{C}^H \mathbf{R}_{yy}^{-1} \mathbf{C})^{-1} \mathbf{f} \quad (2)$$

考虑到 \mathbf{y} 具有各态历经性, \mathbf{R}_{yy} 可以利用 \mathbf{y} 的 N 个观测数据估计得到,比如

$$\mathbf{R}_{yy} \approx \frac{1}{N} \sum_{t=0}^{N-1} \mathbf{y}[t] \mathbf{y}[t]^H \quad (3)$$

式中 $\mathbf{y}[t]$ 为采样时刻 t 对 \mathbf{y} 的观测数据。可以注意到,为了计算式(3),中心节点需要有很大的计算能力(由于协方差矩阵估计及求逆运算量为 $O(M^3)$)。此外,还需要大的通信带宽,特别是在多跳传输模式时。在本文 D-LCMSWF 算法可以避免计算本地局部协方差矩阵估计及求逆运算,以更多的运算量产生最优的 LCMV 波束形成器的输出 $\hat{d} = \hat{\mathbf{w}}^H \mathbf{y}$ 。

2 树型 WSN 中的 D-LCMV 波束形成器

在树型 WSN 的广播网络中,假设每个节点具有给它的每个相邻节点传输不同信号通信链路。定义 z_{kq}^i 为算法中第 i 次迭代期间节点 k 传输给节点 q 的 N 个观测值。信号 z_{kq}^i 融合了节点 k 的传感器信号 \mathbf{y}_k 和从节点 k 的邻居节点(不包括节点 q)获得的所有信号。即

$$z_{kq}^i = \mathbf{w}_k^{iH} \mathbf{y}_k + \sum_{l \in N_k \setminus \{q\}} z_{lk}^i \quad (4)$$

式中: N_k 为节点 k 的邻居节点集; \mathbf{w}_k^i 为 M_k 维融合矢量,同时也作为集中型 LCMV 的解 $\hat{\mathbf{w}}$ 的一部分。与集中式处理不同,D-LCMSWF 波束形成器中的节点不需传输其 M_k 维信号 \mathbf{y}_k ,而每一节点 k 只需传输 M_k 维信号数据压缩/融合形成的单维(标量)信号 $\mathbf{w}_k^{iH} \mathbf{y}_k$ 。以图 1 所示的网络为例,在节点 3 会根据式(4)产生并传输信号 z_{31} , z_{32} 和 z_{34} 。为了避免鸡和蛋的问题,观测量 z_{qk}^i 被排除在求和式(4)之外,更多细节可参考文献[11]发射反馈消除部分。为了提高通信效率,节点 k 可以给邻居节点广播同样的信号 $z_k^i = \mathbf{w}_k^{iH} \mathbf{y}_k + \sum_{l \in N_k} z_{lk}^i$ 。邻居节点 q 可以从 z_k^i 减去它自己发送的信号 z_{qk}^i 来获得信号 z_{kq}^i 。

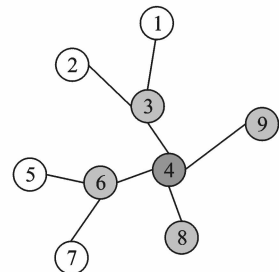


图 1 树型网络拓扑图

Fig. 1 Network graph of tree topology

用 $\mathbf{z}_{\rightarrow k}^i$ 表示所有 \mathbf{z}_{qk}^i 按照要求堆叠的信号矢量, $\forall q \in N_k$ 。如果 $m < n$, 那么 \mathbf{z}_{mk} 要求堆叠在 \mathbf{z}_{nk} 上面。定义

$$\tilde{\mathbf{y}}_k^i = [\mathbf{y}_k^H \quad \mathbf{z}_{\rightarrow k}^{iH}]^H \quad (5)$$

表示节点 k 可获得的所有信号构成的矢量。在 D-LCMSWF 波束形成器中, 每个节点 k 计算一个基于本地输入信号 $\tilde{\mathbf{y}}_k^i$ 的局部 LCMV 波束形成器 $\tilde{\mathbf{w}}_k^i$, 矢量 $\tilde{\mathbf{w}}_k^i$ 可以划分为

$$\tilde{\mathbf{w}}_k^i = [\mathbf{w}_k^H \quad \mathbf{g}_{\rightarrow k}^{iH}]^H \quad (6)$$

这样 $\tilde{\mathbf{w}}_k^{iH} \tilde{\mathbf{y}}_k^i = \mathbf{w}_k^H \mathbf{y}_k + \mathbf{g}_{\rightarrow k}^{iH} \mathbf{z}_{\rightarrow k}^i$ 。定义 \mathbf{w}^i 为所有 \mathbf{w}_k^i 堆叠构成的, 则有

$$\mathbf{w}^i = [\mathbf{w}_1^H \quad \mathbf{w}_2^H \quad \cdots \quad \mathbf{w}_K^H]^H \quad (7)$$

同样, 可以划分定义约束矩阵

$$\mathbf{C} = [\mathbf{C}_1^H \quad \mathbf{C}_2^H \quad \cdots \quad \mathbf{C}_K^H]^H \quad (8)$$

这样 $\mathbf{C}^H \mathbf{w}^i = \sum_{k \in \kappa} \mathbf{C}_k^H \mathbf{w}_k^i$ 。通过引入约束矢量 $\bar{\mathbf{c}}_k^i = \mathbf{C}_k^H \mathbf{w}_k^i$, 定义压缩约束矩阵

$$\bar{\mathbf{C}}^i = [(\mathbf{w}_1^H \mathbf{C}_1)^H \quad (\mathbf{w}_2^H \mathbf{C}_2)^H \quad \cdots \quad (\mathbf{w}_K^H \mathbf{C}_K)^H]^H = [\bar{\mathbf{c}}_1^i \quad \bar{\mathbf{c}}_2^i \quad \cdots \quad \bar{\mathbf{c}}_K^i]^H \quad (9)$$

考虑仅通过 (k, q) 链接进行树型网络图切割, 将网络划分为两个节点集。用 ν_{kq} 表示含有节点 k 的集合, 用 ν_{qk} 表示含有节点 q 的集合。定义

$$\bar{\mathbf{c}}_{kq}^i = \sum_{l \in \nu_{kq}} \bar{\mathbf{c}}_l^i = \sum_{l \in \nu_{kq}} \mathbf{C}_l^H \mathbf{w}_l^i \quad (10)$$

基于与 \mathbf{z}_{qk}^i 在 $\mathbf{z}_{\rightarrow k}^i$ 中同样的排列顺序, 定义 $\bar{\mathbf{C}}_{\rightarrow k}^i$ 表示所有 $\bar{\mathbf{c}}_{qk}^i$ 堆叠的信号矢量, $\forall q \in N_k$ 。基于此, 定义矩阵

$$\mathbf{D}_k^i = [\mathbf{C}_k^H \quad \bar{\mathbf{C}}_{\rightarrow k}^{iH}]^H \quad (11)$$

例如, 考虑如图 1 所示的网络图, 取节点 4 来进行考虑。节点 3 传输给节点 4 的信号等于

$$\mathbf{z}_{34}^i = \mathbf{w}_3^H \mathbf{y}_3 + \mathbf{z}_{13}^i + \mathbf{z}_{23}^i$$

以及

$$\mathbf{z}_{\rightarrow 4}^i = [\mathbf{z}_{34}^{iH} \quad \mathbf{z}_{64}^{iH} \quad \mathbf{z}_{84}^{iH} \quad \mathbf{z}_{94}^{iH}]^H$$

从边 $(3, 4)$ 切割树图, 产生两个节点集 $\nu_{34} = \{1, 2, 3\}$ 和 $\nu_{43} = \{4, 5, 6, 7, 8, 9\}$ 。这样得出

$$\bar{\mathbf{c}}_{34}^i = \bar{\mathbf{c}}_1^i + \bar{\mathbf{c}}_2^i + \bar{\mathbf{c}}_3^i$$

$$\bar{\mathbf{c}}_{43}^i = \bar{\mathbf{c}}_4^i + \bar{\mathbf{c}}_5^i + \bar{\mathbf{c}}_6^i + \bar{\mathbf{c}}_7^i + \bar{\mathbf{c}}_8^i + \bar{\mathbf{c}}_9^i$$

并且, 还可以得到

$$\bar{\mathbf{C}}_{\rightarrow 4}^i = [\bar{\mathbf{c}}_{34}^i \quad \bar{\mathbf{c}}_{64}^i \quad \bar{\mathbf{c}}_{84}^i \quad \bar{\mathbf{c}}_{94}^i]^H =$$

$$[(\bar{\mathbf{c}}_1^i + \bar{\mathbf{c}}_2^i + \bar{\mathbf{c}}_3^i) \quad (\bar{\mathbf{c}}_5^i + \bar{\mathbf{c}}_6^i + \bar{\mathbf{c}}_7^i) \quad \bar{\mathbf{c}}_8^i \quad \bar{\mathbf{c}}_9^i]^H$$

因此树型 WSN 中的 D-LCMV 波束形成器算法步骤如下(为了简化算法描述, 节点 q 执行更新时, 假设 $\bar{\mathbf{C}}_{\rightarrow q}^i$ 总是最新的, 详见文献[15]):

步骤 1 初始化: $i = 0, q = 1$, 并对所有 $\mathbf{w}_k^0, \forall k \in \kappa$ 中的元素初始化为随机数值;

步骤 2 每个节点 $k \in \kappa$, 将其根据式(4)产生 N 个新压缩的传感器观测信号 $\mathbf{z}_{lk}^i [iN + j] = \mathbf{w}_k^H \mathbf{y}_k [iN + j], j = 1, \dots, N, \forall l \in N_k$, 并广播给邻居节点 l ;

步骤 3 每个节点 $k \in \kappa$ 根据这一批观测值计算波束形成器输出信号 d , 计算如下

$$d[iN + j] = \mathbf{w}_k^H \mathbf{y}_k [iN + j] + \sum_{l \in N_k} \mathbf{z}_{lk}^i [iN + j] \quad (12)$$

步骤 4 节点 q 执行如下任务:

(1) 基于 $\tilde{\mathbf{y}}_q^i$ 的 N 个新观测值, 类似于式(3), 重新计算 $\mathbf{R}_{\tilde{\mathbf{y}}_q \tilde{\mathbf{y}}_q}^i = E\{\tilde{\mathbf{y}}_q^i \tilde{\mathbf{y}}_q^{iH}\}$;

(2) 根据 \mathbf{C}_q 和 $\bar{\mathbf{C}}_{\rightarrow q}^i$ 构建 \mathbf{D}_q^i ;

(3) 节点利用公式计算本地 LCMV 波束形成器 $\tilde{\mathbf{w}}_q^{i+1}$ 为

$$\tilde{\mathbf{w}}_q^{i+1} = \begin{bmatrix} \mathbf{w}_q^{i+1} \\ \mathbf{g}_{\rightarrow q}^{i+1} \end{bmatrix} = (\mathbf{R}_{\tilde{\mathbf{y}}_q \tilde{\mathbf{y}}_q}^i)^{-1} \mathbf{D}_q^i (\mathbf{D}_q^{iH} (\mathbf{R}_{\tilde{\mathbf{y}}_q \tilde{\mathbf{y}}_q}^i)^{-1} \mathbf{D}_q^i)^{-1} \mathbf{f} \quad (13)$$

步骤 5 节点 q 将矢量 $\mathbf{g}_{\rightarrow q}^{i+1}$ 的元素对应分发到相应的树枝中(见下一步)。

步骤 6 每个节点 $k \in \kappa \setminus \{q\}$ 更新其 \mathbf{w}_k^i

$$\mathbf{w}_k^{i+1} = g_q^{i+1}(n) \mathbf{w}_k^i \quad (14)$$

式中 $g_q^{i+1}(n)$ 为 $\mathbf{g}_{\rightarrow q}^{i+1}$ 中的第 n 个元素, 且节点 q 含有节点 k 的树枝标签是 n 。

步骤 7 赋值: $i = i + 1, q = (q \bmod K) + 1$; 返回步骤 2。

上述步骤中, 步骤 2 忽略了观测信号 \mathbf{z}_{kq}^i 的一些实际细节。事实上, 构建 \mathbf{z}_{kq}^i 首先是从叶节点向根节点的第一数据流, 紧接着是从根节点流向叶节点的第二数据流。如需详细了解树型拓扑网络结构信号融合的数据流请参考文献[11]。

3 树型 WSN 中 D-LCMSWF 波束形成器

本文提出的树型 WSN 中的 D-LCMSWF 波束形成器是将 D-LCMV 步骤 4 中式(13)的权值 $\tilde{\mathbf{w}}_k^i$ 的计算, 通过正交投影转化为线性约束和自适应滤波两个部分, 然后在自适应滤波部分成功采用 MSWF 方法进行等效替代来降低运算量。MSWF 方法不需要估计协方差矩阵, 不需要协方差矩阵的特征值分解, 计算复杂度低, 收敛速度比最小均方 (Least mean square, LMS) 算法快。MSWF 方法代表了降秩自适应滤波的最新研究成果, 是一种新的信号处理工具[19]。

下面详细介绍利用 MSWF 方法隐式计算步骤 4 中式(13)的计算方法。式(13)权值 $\tilde{\mathbf{w}}_q^{i+1}$ 的计算

采用 MSWF 方法进行等效替代,处理流程如图 2 所示(以 4 级 MSWF 为例),为了简化将 MSWF 处理图中符号的迭代上标 i 省略。由图可知,通过将信号 $\tilde{\mathbf{y}}_q^i$ 进行约束正交投影(其中 $\mathbf{B}\mathbf{w}_f = \mathbf{0}$)来将 LCMV 问题转化为广义滤波器问题,划分为线性约束和自适应滤波两部分,并借助于 MSWF 方法来解决,进而避免直接采用式(13)计算 LCMV 的计算量大的问题。这里线性约束部分 $\mathbf{w}_f^H \mathbf{D}_q^i = \mathbf{f}^H$

可保证算法与 LCMV 等效,并可求得 $\mathbf{w}_f = \mathbf{D}_q^i (\mathbf{D}_q^{iH} \mathbf{D}_q^i)^{-1} \mathbf{f}$,且容易得到 $\mathbf{B}\mathbf{D}_q^i = \mathbf{0}$ 。对于阻塞矩阵 \mathbf{B} 的计算,可通过对 \mathbf{D}_q^i 奇异值分解得到 $\mathbf{D}_q^i = \mathbf{U}\mathbf{A}\mathbf{V}^H$,则 \mathbf{B} 为矩阵 \mathbf{U} 删去前 Q 列后所得矩阵的共轭转置。图 2 中,由 $d_0 = \mathbf{w}_f^H \tilde{\mathbf{y}}_q^i$ 和 $\mathbf{X}_0 = \mathbf{B}\tilde{\mathbf{y}}_q^i$ 可计算得到投影后数据 d_0 和 \mathbf{X}_0 ,然后将此两部分数据送入到 MSWF 中进行处理。图 2 中,矩形虚线框表示的自适应滤波器权值关系为 $\mathbf{W}_{X_0} = \mathbf{T}\mathbf{W}_Z^{[19]}$ 。

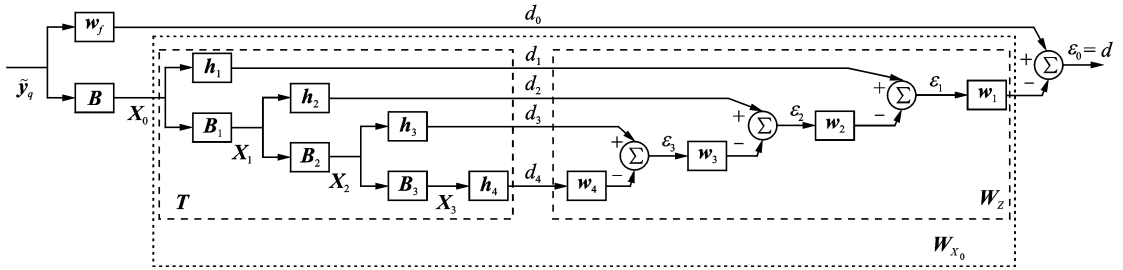


图 2 节点 q 的 D-LCMSWF 算法处理图

Fig. 2 Flow chart of D-LCMSWF algorithm for node q

MSWF 算法的具体步骤如下:

步骤 1 初始化:得到 d_0 和 \mathbf{X}_0 ;

步骤 2 前项递推:For $i' = 1, 2, \dots, D$

$$\begin{aligned} \mathbf{h}_{i'} &= E[d_{i'-1}^* \mathbf{X}_{i'-1}] / \|E[d_{i'-1}^* \mathbf{X}_{i'-1}]\|_2; \\ d_{i'} &= \mathbf{h}_{i'}^H \mathbf{X}_{i'-1}; \\ \mathbf{B}_{i'} &= \text{null}\{\mathbf{h}_{i'}\} = \mathbf{I}_M - \mathbf{h}_{i'} \mathbf{h}_{i'}^H; \\ \mathbf{X}_{i'} &= \mathbf{B}_{i'} \mathbf{X}_{i'-1}; \end{aligned}$$

步骤 3 后项递推:

$$\begin{aligned} \epsilon_D &= d_D, \text{For } i' = D, D-1, \dots, 1 \\ \omega_{i'} &= E[d_{i'-1}^* \epsilon_{i'}] / E[|\epsilon_{i'}|^2]; \\ \epsilon_{i'-1} &= d_{i'-1} - \omega_{i'}^* \epsilon_{i'}. \end{aligned}$$

其中 D 表示递推级数;算法中符号 $\|\cdot\|_2$ 表示 2 范数;上标 $*$ 表示共轭计算。

很容易知道 $\|\mathbf{h}_{i'}\|_2 = 1$, $\mathbf{B}_{i'} \mathbf{h}_{i'} = \mathbf{0}$, 上面算法中 $\mathbf{B}_{i'}$ 满足阻塞矩阵条件。

通过计算可知,图 2 中 \mathbf{W}_Z 表示为

$$\mathbf{W}_Z = \begin{bmatrix} \omega_1 & -\omega_1 \omega_2 & \cdots & (-1)^{D+1} \prod_{i'=1}^D \omega_{i'} \end{bmatrix}^T \quad (15)$$

通过计算,在图 2 中的满秩分解预处理算子 \mathbf{T} 可以表示为

$$\mathbf{T} = \begin{bmatrix} \mathbf{h}_1 & \mathbf{B}_1^H \mathbf{h}_2 & \cdots & \prod_{i'=1}^{D-2} \mathbf{B}_{i'}^H \mathbf{h}_{D-1} & \prod_{i'=1}^{D-1} \mathbf{B}_{i'}^H \mathbf{h}_D \end{bmatrix} \quad (16)$$

根据 $\mathbf{W}_{X_0} = \mathbf{T}\mathbf{W}_Z$, 以及结合文献[17, 19, 20], 推得

式(13)的 MSWF 等效计算式为

$$\tilde{\mathbf{w}}_q^{i+1} = \mathbf{w}_f - \mathbf{B}^H \mathbf{W}_{X_0} = \mathbf{w}_f - \mathbf{B}^H (\mathbf{T}\mathbf{W}_Z) \quad (17)$$

在 MSWF 算法中,计算阻塞矩阵 $\mathbf{B}_{i'} = \text{null}\{\mathbf{h}_{i'}\}$ 所需要的运算量相当大。因此 MSWF 算法还可以采用基于相关相减结构的多级维纳滤波器^[19-20] (MSWF using the correlation subtraction algorithm, CSA-MSWF)来避免构造阻塞矩阵。CSA-MSWF 与 MSWF 算法功能相同,但运算量更小,其算法结构可以直接引入,关于 CSA-MSWF 算法更多内容请参阅文献[19, 20],不再赘述。

需要注意,文献[15]中定理 IV. 1 给出树型网络中最优充分条件只有当 $Q < 2$ 时能够满足。与约束条件数 Q 相比,那些有小数量的邻居节点的节点(比如叶节点)常出现处于非理想平衡会情况。然而,对于次优平衡点可以通过监测是否 \bar{C}^i 的第 Q 个大奇异值是否趋于零(即 $\sigma_Q(\bar{C}^i) \rightarrow 0$)来判定,然后采取追溯措施排除它。即,节点 q 可以将本地的 \mathbf{w}_q^i 分为两个线性独立分量

$$\mathbf{w}_q^i = \mathbf{w}_{q,1}^i + \mathbf{w}_{q,2}^i \quad (18)$$

并传播 \mathbf{z}_q^i 信号的两个部分,就像由两个不同的(虚拟)节点产生,即, $\mathbf{z}_{q,1}^i = \mathbf{w}_{q,1}^{iH} \mathbf{y}_q$, $\mathbf{z}_{q,2}^i = \mathbf{w}_{q,2}^{iH} \mathbf{y}_q$ 。每当 $\sigma_Q(\bar{C}^i) \rightarrow 0$ 时,就可以执行这样的修正,直到收敛到 $\hat{\mathbf{w}}$ 。一旦不理想的点消失了,这两个本地的滤波器 $\mathbf{w}_{q,1}^i$ 和 $\mathbf{w}_{q,2}^i$ 可以再次合并,传输一个信号 \mathbf{z}_q^i 。

4 计算量分析

假设信号 \mathbf{X}_0 维数为 P , 样本量为 N (即 \mathbf{X}_0 为

$P \times N$ 复信号矩阵), 线性约束条件数为 Q , 则 $\hat{\mathbf{y}}_k^i$ 为 $(P+Q) \times N$ 复信号矩阵。树形 WSN 中, 采用 D-LCMSWF 相比 D-LCMV 直接计算式 (13) 来说, 避免了协方差矩阵 $\mathbf{R}_{\mathbf{y}_q^i \mathbf{y}_q^i}$ 估计的运算量 $O((P+Q)^2 N)$ 及协方差矩阵 $\mathbf{R}_{\mathbf{y}_q^i \mathbf{y}_q^i}$ 求逆的运算量 $O((P+Q)^3)$, 因此每个节点可节约复乘运算量约为 $O((P+Q)^3 + (P+Q)^2 N)$ 。虽然相比 D-LCMV 来说, D-LCMSWF 约束正交投影部分另外需要进行 \mathbf{D}_k^i 的奇异值分解, 按照文献 [21] 提供的方式精确计算 $\mathbf{D}_k^i = \mathbf{U} \mathbf{\Lambda} \mathbf{V}^H$ 中的矩阵 \mathbf{U} , 计算复杂度约为 $10Q^3$ 。一般来说 $Q \ll P \ll N$, 故而 \mathbf{D}_k^i 奇异值分解的计算量相比 $O((P+Q)^3 + (P+Q)^2 N)$ 来说可以忽略。因此树形 WSN 中采用 D-LCMSWF 比 D-LCMV 可节约运算量约为 $O((P+Q)^3 + (P+Q)^2 N)$ 。

下面再分析当 D-LCMSWF 算法采用 MSWF 与 CSA-MSWF 结构来实现时的计算量对比。利用 MSWF 计算 $\mathbf{X}_i = \mathbf{B}_i \mathbf{X}_{i-1}$ 需要的复数乘法计算量为 $O(P^2 N)$, 而在 CSA-MSWF 结构中, 利用 $\mathbf{X}_i = \mathbf{X}_{i-1} - \mathbf{h}_i d_i$ 来替代计算 $\mathbf{X}_i = \mathbf{B}_i \mathbf{X}_{i-1}$ 需要的计算量仅为 $O(PN)$ 。采用 CSA 结构后, 由于 MSWF 的每一级均可得到从 $O(P^2 N)$ 到 $O(PN)$ 的降低, 因此采用 CSA 结构可有效降低 MSWF 的计算量。另外, 可根据实际应用设定一个较小门限值, 当 MSWF 算法中第 i' 级的输出 d_i 小于门限时, 可对 MSWF 截断处理, 从而在算法达到稳定时大量节约节点的运算量。

5 仿真分析

对 D-LCMSWF 在树型 WSN 网络进行蒙特卡罗 (Monte-Carlo, MC) 仿真, 并与 D-LCMV 波束形成器进行比较。在每次 MC 实验中, 重新构造 20 维随机变量 \mathbf{x} 表示时间、空间上统计独立, 服从区间 $[-0.5, 0.5]$ 上均匀分布的本地信号源。这 20 个信号源由 K 个节点 WSN 进行观测, 每个节点有 $M_k = 6$ 个传感器 (整个网络具有 $M = 6K$ 个传感器)。由信号源到传感器形成 20 个 M 维导向矢量, 这些导向矢量定义为 $M \times 20$ 方向矩阵 \mathbf{A} 的列矢量。假定 \mathbf{A} 的各个元素服从区间 $[-0.5, 0.5]$ 上的均匀分布。这 M 个传感器在采样时刻 t 收集的观测数据, 表示为

$$\mathbf{y}[t] = \mathbf{A} \mathbf{x}[t] + \mathbf{n}[t] \quad (19)$$

式中: $\mathbf{x}[t]$ 为随机变量 \mathbf{x} 在 t 时刻的样本; $\mathbf{n}[t]$ 为在时间上服从零均值均匀分布, 且在空间不相关的噪声观测矢量, 假设其功率是随机变量 \mathbf{x} 中信号源功

率的一半。

为了评估分布式波束形成器的性能, 在不同的迭代之间计算分布式波束形成器和相应的集中型 LCMV 波束形成器的输出信噪比 (SNR) 之间的差异 (单位 dB)。即

$$\Delta \text{SNR}^i = 10 \log(\hat{\mathbf{w}}^H \mathbf{R}_{\mathbf{y}_i} \hat{\mathbf{w}}) - 10 \log(E(|d^i|^2)) \quad (20)$$

理想的情况下, 这种差异收敛到 $\Delta \text{SNR}^i \rightarrow 0$ dB。此外, 还计算在不同迭代之间, 分布式与集中型 LCMV 波束形成器生成的权值的均方差。即

$$\frac{1}{M} \|\mathbf{w}^i - \hat{\mathbf{w}}\| \quad (21)$$

在树型无线传感器网络环境, 基于式 (19) 产生传感信号模型, 对 D-LCMSWF 和 D-LCMV 波束形成器进行仿真。约束矩阵 \mathbf{C} 总是选择方向矩阵 \mathbf{A} 的前 Q 列构成, 响应向量表示为 $\mathbf{f} = [1 \ \dots \ 1]^T$ 。这相当于期望输出响应是矩阵 \mathbf{A} 中前 Q 个信号源的求和。相关矩阵基于 $N = 1\ 000$ 个样本的时间平均来估计, 这些样本在每次迭代中被循环使用。对于每次选定的 K 和 Q 执行 30 次的 MC 仿真, 并利用所得结果的中位数来绘制性能曲线。

对于 $K = 12$ 个节点情况下, D-LCMSWF 与集中型 LCMV 波束形成器的输出 SNR 差异以及权值系数均方差对比仿真结果分别如图 3(a, b) 所示。图 3(a, b) 中分别显示 $Q = 1, 2, 3, 4$ 情况下式 (20, 21) 的仿真结果。从图 3(a) 中可以看到, 随着 Q 值变大, 收敛速度逐渐减小。从图 3(b) 中可以看到, 当 $Q > 1$ 时, 算法不能够保证收敛到最优解; 当 $Q = 1$ 时, 算法总能够收敛到最优解。

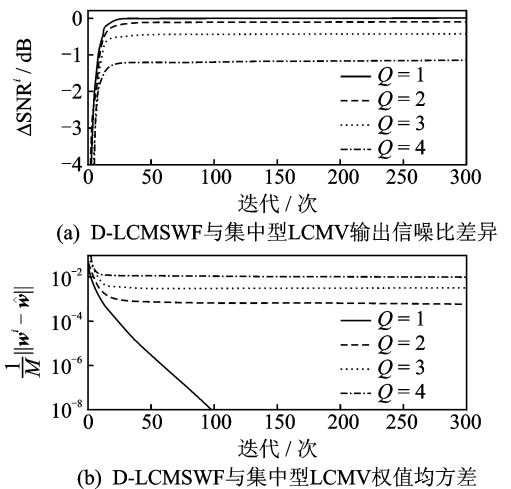
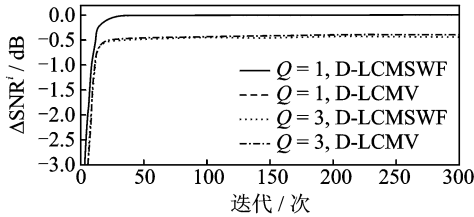


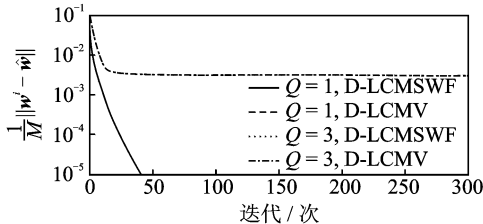
图 3 D-LCMSWF 与集中型 LCMV 输出信噪比差异及权值均方差

Fig. 3 Difference in output SNR and mean square error between D-LCMSWF and centralized LCMV

图 4(a, b) 给出了 $K=12, N=1\ 000, Q=1, 3$ 约束时, 树型网络中 D-LCMSWF 和 D-LCMV 波束形成器的输出结果与集中型 LCMV 的对比。从图 4 中可以看出在不同线性约束条件 Q 下, D-LCMSWF 和 D-LCMV 的各自对应的输出结果基本重合。故而验证了 D-LCMV 和 D-LCMSWF 波束形成器的等效性。



(a) D-LCMSWF及D-LCMV与集中型LCMV输出信噪比差异

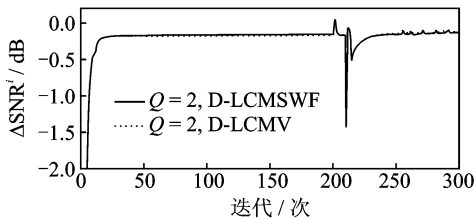


(b) D-LCMSWF及D-LCMV与集中型LCMV权值均方差

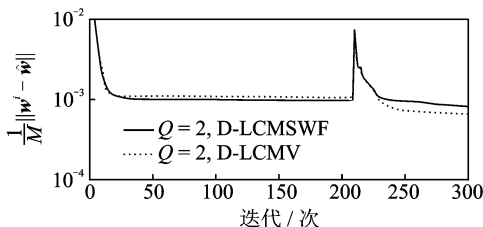
图 4 D-LCMSWF 与 D-LCMV 的输出对比 ($K=12$)

Fig. 4 Output comparison of D-LCMSWF and D-LCMV ($K=12$)

图 5(a, b) 给出了 $K=10, N=1\ 000, Q=2$ 约束时树型网络中 D-LCMSWF 和 D-LCMV 波束形成器的输出结果与集中型 LCMV 的对比。从图 5 中可见 D-LCMSWF 和 D-LCMV 波束形成器的输出结果基本重合, 进一步验证了两算法等效性。在实验中, D-LCMSWF 和 D-LCMV 算法收敛于局



(a) D-LCMSWF及D-LCMV与集中型LCMV输出信噪比差异



(b) D-LCMSWF及D-LCMV与集中型LCMV权值均方差

图 5 D-LCMSWF 与 D-LCMV 的输出对比 ($K=10$)

Fig. 5 Output comparison of D-LCMSWF and D-LCMV ($K=10$)

部最优点时, 会采用修正算法来使两算法脱离局部最优点的困扰。图 5 中显示, 实验中在迭代到 200 次时(和迭代 200 次之后)采用修正算法避免收敛于局部最优解。图 5 中 $i=200$ 处于局部最优点, 进行修正后权值出现波动, 之后权值开始再次继续向最优解收敛。

6 结束语

本文研究了 D-LCMSWF 在树型拓扑结构网络中的波束形成算法。此算法由于避免协方差矩阵估计及求逆运算, 在达到与 D-LCMV 相同输出性能情况下, 节约运算量约为 $O((P+Q)^3 + (P+Q)^2 N)$ 。另外采用 CSA-MSWF 时, 每一级均可得到从 $O(P^2 N)$ 到 $O(PN)$ 的降低。例如在一个树形 WSN 中, 某一节点 k 邻居节点数为 10 个(包含节点 k 自身), 设每节点传感器数 $M_k=6$, 样本数 $N=1\ 000$, 线性约束数 $Q=2$, 则 $\tilde{\mathbf{y}}_k^i$ 为 $(9+6) \times 1\ 000$ 复信号矩阵; 因此, 此节点采用 D-LCMSWF 比 D-LCMV 在每次迭代所节约运算量约为 $O((9+6)^3 + (9+6)^2 \times 1\ 000) = O(3\ 375 + 225\ 000)$ 。设整个 WSN 网络迭代计算次数 $i=200$, 若其中节点 k 执行了 20 次迭代计算, 则此节点 k 节约运算量为 $O(20 \times (3\ 375 + 225\ 000))$; 而在进行 D-LCMSWF 算法前要对 \mathbf{D}_k^i 奇异值分解, 所需计算量约为 $10 \times (2)^3 = 80$, 因此可忽略不计。此节点若采用 CSA-MSWF 结构相比 MSWF 结构每一级还可得到从 $O(9^2 \times 1\ 000)$ 到 $O(9 \times 1\ 000)$ 计算量降低, 即每一级节约运算量 $O(72\ 000)$; 以满秩分解为例, 每次迭代需要 9 级分解, CSA-MSWF 相比 MSWF 可节约运算量 $O(9 \times 72\ 000)$; 设整个 WSN 网络迭代计算次数 $i=200$, 若其中节点 k 执行了 20 次迭代计算, 则此节点 k 采用 CSA-MSWF 相比 MSWF 节约运算量为 $O(20 \times (9 \times 72\ 000))$ 。此外, 在 MSWF 算法达到稳定状态时可以通过截断处理大量节约节点的运算量。

参考文献:

- [1] Fang J, Li H. Optimal/near-optimal dimensionality reduction for distributed estimation in homogeneous and certain inhomogeneous scenarios [J]. IEEE Transactions on Signal Processing, 2010, 58(8): 4339-4353.
- [2] Bertrand A, Moonen M. Seeing the bigger picture: How nodes can learn their place within a complex ad hoc network topology [J]. IEEE Signal Processing

- Magazine, 2013, 30(3): 71-82.
- [3] Bertrand A, Moonen M. Distributed computation of the Fiedler vector with application to topology inference in ad hoc networks [J]. *Signal Processing*, 2013, 93(5): 1106-1117.
- [4] Salman N, Ghogho M, Kemp A H. Optimized low complexity sensor node positioning in wireless sensor networks [J]. *IEEE Sensors Journal*, 2014, 14(1): 39-46.
- [5] Bejar Haro B, Zazo S, Palomar D P. Energy efficient collaborative beamforming in wireless sensor networks [J]. *IEEE Transactions on Signal Processing*, 2014, 62(2): 496-510.
- [6] Cattivelli F S, Sayed A H. Diffusion LMS strategies for distributed estimation [J]. *IEEE Transactions on Signal Processing*, 2010, 58(5): 1035-1048.
- [7] Bertrand A, Moonen M, Sayed A H. Diffusion bias-compensated RLS estimation over adaptive networks [J]. *IEEE Transactions on Signal Processing*, 2011, 59(11): 5212-5224.
- [8] Bertrand A, Moonen M. Consensus-based distributed total least squares estimation in ad hoc wireless sensor networks [J]. *IEEE Transactions on Signal Processing*, 2011, 59(5): 2320-2330.
- [9] Bertrand A, Moonen M. Distributed adaptive node-specific signal estimation in fully connected sensor networks—Part I: Sequential node updating [J]. *IEEE Transactions on Signal Processing*, 2010, 58(10): 5277-5291.
- [10] Bertrand A, Szurley J, Ruckebusch P, et al. Efficient calculation of sensor utility and sensor removal in wireless sensor networks for adaptive signal estimation and beamforming [J]. *IEEE Transactions on Signal Processing*, 2012, 60(11): 5857-5869.
- [11] Bertrand A, Moonen M. Distributed adaptive estimation of node specific signals in wireless sensor networks with a tree topology [J]. *IEEE Transactions on Signal Processing*, 2011, 59(5): 2196-2210.
- [12] Szurley J, Bertrand A, Ruckebusch P, et al. Greedy distributed node selection for node-specific signal estimation in wireless sensor networks [J]. *Signal Processing*, 2014, 94(1): 57-73.
- [13] Bertrand A, Moonen M. Distributed node-specific LCMV beamforming in wireless sensor networks [J]. *IEEE Transactions on Signal Processing*, 2012, 60(1): 233-246.
- [14] Himawan I, McCowan I, Sridharan S. Clustered blind beamforming from ad-hoc microphone arrays [J]. *IEEE Transactions on Audio, Speech, Language Processing*, 2011, 19(4): 661-676.
- [15] Bertrand A, Moonen M. Distributed LCMV beamforming in a wireless sensor network with single-channel per-node signal transmission [J]. *IEEE Transactions on Signal Processing*, 2013, 61(13): 3447-3459.
- [16] 黄刚, 王汝传, 许一帆. 无线传感器网络中基于分簇广播认证协议方案 [J]. *南京航空航天大学学报*, 2010, 42(1): 72-76.
Huang Gang, Wang Ruchuan, Xu Yifan. Broadcast authentication protocol scheme based on layer-cluster in WSN [J]. *Journal of Nanjing University of Aeronautics & Astronautics*, 2010, 42(1): 72-76.
- [17] Goldstein J S, Reed I S, Scharf L L. A multistage representation of the wiener filter based on orthogonal projections [J]. *IEEE Transactions on Information Theory*, 1998, 44(7): 2943-2959.
- [18] 张林让, 廖桂生, 罗丰. 有限次快拍下自适应波束形成 [J]. *西安电子科技大学学报*, 2000, 27(5): 542-545.
Zhang Linrang, Liao Guisheng, Luo Feng. Adaptive beamforming via finite snapshots [J]. *Journal of Xidian University*, 2000, 27(5): 542-545.
- [19] 王永良, 丁前军, 李荣锋. 自适应阵列处理 [M]. 北京: 清华大学出版社, 2009.
Wang Yongliang, Ding Qianjun, Li Rongfeng. *Adaptive array processing* [M]. Beijing: Tsinghua University Press, 2009.
- [20] 黄庆东. 最小模级联相消器算法研究 [D]. 西安: 西安电子科技大学, 2011.
Huang Qingdong. Research on minimum module cascaded canceller algorithm [D]. Xi'an: Xidian University, 2011.
- [21] Barlow J. More accurate bidiagonal reduction for computing the singular value decomposition [J]. *SIAM Journal on Matrix Analysis and Applications*, 2002, 23(3): 761-798.

