BEAM COORDINATION VIA DIFFUSION ADAPTATION OVER ARRAY NETWORKS

Xiaochuan Zhao, Jianshu Chen, and Ali H. Sayed

Department of Electrical Engineering
University of California, Los Angeles

ABSTRACT

In this work, we consider a distributed beam coordination problem, where a collection of arrays are interconnected by a certain topology. The beamformers employ an adaptive diffusion strategy to compute the beamforming weight vectors by relying solely on cooperation with their local neighbors. We analyze the mean-square-error (MSE) performance of the proposed strategy, including its transient and steady-state behavior. Simulation results support the findings that the MSE performance improves uniformly across the network relative to non-cooperative designs.

***Index Terms*—** Beam coordination, distributed beamforming, diffusion adaptation, diagonal loading, affine constraints.

1. INTRODUCTION

Prior work in the literature has shown that it is beneficial to deploy distributed cooperation strategies to coordinate the operation of beamforming arrays [1] or antenna elements [2–6] that are scattered over a geographical area and interconnected via some topology. Along these lines, in this work, we develop adaptive diffusion strategies to coordinate the beampatterns of arrays in a distributed manner. Our approach is different from the literature [1–6] in a couple of aspects. First, each individual array forms a beampattern pointing towards the desired signal. Arrays suffering from noise levels improve their beampatterns through cooperation with their neighboring arrays. Second, the diffusion strategy is fully distributed; no fusion center or master node is required to compute the beamforming weighting vectors for the arrays. Third, the arrays in the network do not need to enforce synchronization of their carrier phases. Fourth, the diffusion strategy does not require the arrays to share raw data. Besides, the diffusion strategy is iterative and computationally efficient; no complex matrix inversions are required.

Diffusion strategies, proposed in [7, 8], are scalable and robust, and endow networks with real-time adaptation and learning abilities. They have been applied before to model various forms of complex and self-organized behavior in biological networks [9, 10] and to the solution of general optimization problems [11].

In this article, we formulate a constrained version of diffusion strategies and apply it to coordinate the beampatterns of arrays distributed over an area and connected via a topology. In the sequel, we first fit different types of beamforming problems into a general optimization framework, which is subsequently solved by means of a diffusion strategy with regularization and affine constraints. We analyze the mean-square convergence conditions and steady-state performance. Simulation results illustrate that the diffusion strategy improves the beamforming performance of array networks over the non-cooperative case.

This work was supported in part by NSF grants CCF-0942936 and CCF-1011918. Email: {xzhao, jshchen, sayed}@ee.ucla.edu.

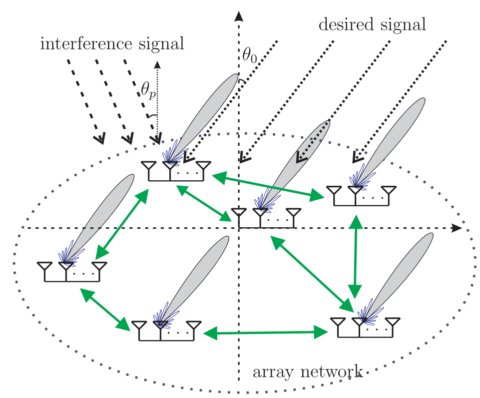


Fig. 1. An illustration of a collection of interconnected arrays.

Notation: We use lowercase letters to denote vectors, uppercase letters for matrices, plain letters for deterministic variables, and boldface letters for random variables. We also use $(\cdot)^*$ to denote conjugate transposition, $(\cdot)^{-1}$ for matrix inversion, $\mathrm{Tr}(\cdot)$ for the trace of a matrix, $\otimes$ for Kronecker products, $\rho(\cdot)$ for the spectral radius of a matrix, and $\mathrm{vec}(\cdot)$ for a vector constructed by stacking the columns of a matrix. All vectors in our treatment are column vectors, with the exception of the regression vectors, $\boldsymbol{u}_{k,i}$, which are taken to be row vectors for convenience of presentation.

2. PROBLEM FORMULATION

Consider a network of N identical antenna arrays, as shown in Fig. 1. Each array consists of M antenna elements that are receiving a desired narrow-band signal $\boldsymbol{s}_0(i) \in \mathbb{C}$ from a far field, while suffering interference from some other signals $\boldsymbol{s}_1(i), \dots, \boldsymbol{s}_P(i) \in \mathbb{C}$, $P \geq 0$, at the same time. The received signal at the kth array is therefore modeled as:

$$\boldsymbol{u}_{k,i}^* = a(\theta_0)\boldsymbol{s}_0(i) + \underbrace{\sum_{p=1}^{P} \boldsymbol{s}_p(i)a(\theta_p) + \boldsymbol{n}_{k,i}}_{\boldsymbol{z}_{k,i}} \tag{1}$$

where $\boldsymbol{u}_{k,i}^*$ is an $M \times 1$ vector that collects the received signals at array k, $\{a(\theta_0), a(\theta_1), \dots, a(\theta_P)\}$ are $M \times 1$ array manifold vectors (steering vectors) [12] for the desired and interference signals, $\{\theta_0, \theta_1, \dots, \theta_P\}$ are the corresponding directions-of-arrival (DOA), and $\boldsymbol{n}_{k,i}$ is the additive noise vector at array k and time i. For a

978-1-4673-0971-4/12/$31.00 © 2012 IEEE

uniform linear array (ULA), the array manifold vector $a(\theta)$ has the following form:

$$a(\theta) = \mathrm{col}\big\{1,\, e^{-j\frac{2\pi\Delta}{\lambda_c}\sin\theta},\, \cdots,\, e^{-j\frac{2\pi(M-1)\Delta}{\lambda_c}\sin\theta}\big\} \quad (2)$$

where Δ denotes the spacing between two adjacent antenna elements in each array, and λ_c denotes the wavelength of the carrier signal. We aggregate the interference and noise signals into a perturbation vector, $\boldsymbol{z}_{k,i}$. It is the statistical profile of $\boldsymbol{z}_{k,i}$ that matters and it will be learned by the adaptive algorithm in real-time.

Beamforming problems generally deal with the design of a weight vector $w \in \mathbb{C}^{M\times 1}$ in order to recover the desired signal $\boldsymbol{s}_0(n)$ from the received data $\boldsymbol{u}_{k,i}$. The output of the beamformer is a weighted combination of the following form:

$$\hat{\boldsymbol{s}}_0^*(i) = \boldsymbol{u}_{k,i} w \quad (3)$$

Many design criteria can be used to design the individual beamformers, such as minimum-variance-distortionless-response design (MVDR), linearly-constrained-minimum-variance design (LCMV), minimum-mean-square-error design (MMSE), and maximum-signal-to-interference-plus-noise-ratio design (MSINR). The main purpose of all these criteria is to suppress the influence of the perturbation component on the output of the beamformer while preserving the signal component. Several of these criteria can be viewed as special cases of the following formulation:

$$\begin{aligned} &\underset{w}{\text{minimize}} \quad \mathbb{E}|\boldsymbol{d}_k(i) - \boldsymbol{u}_{k,i}w|^2 \\ &\text{subject to} \qquad G_k^* w = b_k \end{aligned} \quad (4)$$

where G_k is an $M \times L_k$ matrix and b_k is an $L_k \times 1$ vector. The affine constraint is used to specify the desired response of the beamformer to certain directions. For example, along the directions that have clutter or jammers, we may assign small response values by choosing the columns of G_k to be the array manifold vectors in these directions and setting the corresponding entries of b_k to small values. If we know the direction of the desired signal beforehand, then we can set unit response to that direction so that the desired signal passes through the array without distortion. We summarize the different choices for $\{\boldsymbol{d}_k(i), G_k, b_k\}$ for different criteria in Table 1. We note that it was shown in [12, pp. 439–452] that beamformers obtained under the criteria MVDR, MMSE, and MSINR are equivalent up to some scaling for narrow-band signals.

Table 1. Parameter selections for various design criteria.

	$\boldsymbol{d}_k(i)$	G_k	b_k
MMSE	$\boldsymbol{s}_0^*(i)$	0	0
LCMV	0	G_k	b_k
MVDR	0	$a(\theta_0)$	1

For the array network in Fig. 1, our objective is to find an optimal weight vector w that minimizes the following *global* cost function, which integrates the performance across all arrays and includes a regularization term:

$$\begin{aligned} &\underset{w}{\text{minimize}} \;\; J^{\text{glob}}(w) \triangleq w^*\Pi w + \sum_{k=1}^{N} \mathbb{E}|\boldsymbol{d}_k(i) - \boldsymbol{u}_{k,i}w|^2 \\ &\text{subject to} \qquad G^* w = b \end{aligned} \quad (5)$$

where Π is an $M \times M$ positive semi-definite weighting matrix. The regularization term $w^*\Pi w$ is used to discourage solutions with large power or to prevent rank deficiency. In array processing, the regularization term corresponds to a diagonal loading technique that improves robustness [13]. Furthermore, the matrices G and b are defined as

$$G \triangleq \mathrm{row}\{G_1,\, G_2,\, \ldots,\, G_{N'}\}, \quad b \triangleq \mathrm{col}\{b_1,\, b_2,\, \ldots,\, b_{N'}\} \quad (6)$$

where N' is the number of effective constraints across the network; we merge different pairs of $\{G_k, b_k\}$ into one if they represent equivalent constraints. The matrix G has dimensions $M \times L$ and is assumed to be full rank with $L < M$.

3. DIFFUSION STRATEGIES

In our problem setup, w^o could be directly solved from (4) if the individual arrays know the data statistics beforehand. When this information is missing, an adaptive solution becomes desirable to enable the arrays to rely on the instantaneous data. Extending the arguments of [8, 11] to constrained estimation problems, we can derive the following distributed algorithm for adapting the weight vector $\boldsymbol{w}_{k,i}$ at the various arrays. For each array k, the algorithm involves two steps. The first step uses data $\{\boldsymbol{d}_k(i), \boldsymbol{u}_{k,i}\}$ that are available at array k at time i to update the existing beamforming vector estimate $\boldsymbol{w}_{k,i-1}$ to an intermediate value $\boldsymbol{\psi}_{k,i}$. The second step aggregates the intermediate estimates $\{\boldsymbol{\psi}_{l,i}\}$ from the neighbors of array k to obtain the updated estimate $\boldsymbol{w}_{k,i}$. All arrays in the network perform similar steps simultaneously. This form of the algorithm is known as the adapt-then-combine (ATC) diffusion strategy. It is described by the equations:

$$\begin{cases} \boldsymbol{\phi}_{k,i} = (I_M - \mu_k\Pi)\boldsymbol{w}_{k,i-1} + \mu_k \boldsymbol{u}_{k,i}^*(\boldsymbol{d}_k(i) - \boldsymbol{u}_{k,i}\boldsymbol{w}_{k,i-1}) \\ \boldsymbol{\psi}_{k,i} = P_G^{\perp}\boldsymbol{\phi}_{k,i} + G(G^*G)^{-1}b \\ \boldsymbol{w}_{k,i} = \displaystyle\sum_{l\in\mathcal{N}_k} a_{lk}\boldsymbol{\psi}_{l,i} \end{cases} \quad (7)$$

where the first step updates the estimates based on the new data, the second step projects the estimates in order to obtain feasible solutions [14], and the third step fuses information from neighboring arrays. The orthogonal projection matrix $P_G^{\perp}$ is given by

$$P_G^{\perp} \triangleq I - G(G^*G)^{-1}G^* \quad (8)$$

The combination coefficients $\{a_{lk}\}$ satisfy

$$\sum_{l\in\mathcal{N}_k} a_{lk} = 1, \quad a_{lk} \geq 0, \quad a_{lk} = 0 \;\text{ if }\; l \notin \mathcal{N}_k \quad (9)$$

where $\mathcal{N}_k$ denotes the set of neighbors of array k. We collect $\{a_{lk}\}$ into an $N \times N$ left-stochastic matrix A; it satisfies $A^{\mathsf{T}}\mathbb{1}_N = \mathbb{1}_N$. The ATC diffusion solution (7) is feasible because

$$\begin{aligned} G^*\boldsymbol{w}_{k,i} &= \sum_{l\in\mathcal{N}_k} a_{lk}G^*\boldsymbol{\psi}_{l,i} \\ &= \sum_{l\in\mathcal{N}_k} a_{lk}G^*P_G^{\perp}\boldsymbol{\phi}_{l,i} + \sum_{l\in\mathcal{N}_k} a_{lk}G^*G(G^*G)^{-1}b \\ &= \left(\sum_{l\in\mathcal{N}_k} a_{lk}\right) b \\ &= b \end{aligned} \quad (10)$$

for $k = 1, 2, \ldots, N$.

3.1. Error Recursions

Since all arrays are interested in converging towards the same beamforming vector, it is reasonable to assume that there exists a vector w^o that relates the data $\{\boldsymbol{d}_k(i), \boldsymbol{u}_{k,i}\}$ up to some perturbation, namely, that

$$\boldsymbol{d}_k(i) = \boldsymbol{u}_{k,i} w^o + \boldsymbol{v}_k(i) \tag{11}$$

where $\boldsymbol{v}_k(i)$ denotes some zero-mean measurement noise at time i and is independent of all other random variables. The $M \times 1$ vector w^o denotes the parameter of interest that problem (5) is attempting to estimate; it also satisfies the affine constraint $G^* w^o = b$.

Introduce the error vectors

$$\widetilde{\boldsymbol{w}}_{k,i} \triangleq w^o - \boldsymbol{w}_{k,i} \tag{12}$$

and let

$$\boldsymbol{R}_{k,i} \triangleq \boldsymbol{u}_{k,i}^* \boldsymbol{u}_{k,i} \tag{13}$$

$$\boldsymbol{s}_{k,i} \triangleq \boldsymbol{u}_{k,i}^* \boldsymbol{v}_k(i) \tag{14}$$

We collect various quantities across all nodes in the network into the following global block vector and matrix quantities:

$$\boldsymbol{\mathcal{R}}_i \triangleq \text{diag}\{\boldsymbol{R}_{1,i}, \ldots, \boldsymbol{R}_{N,i}\} \tag{15}$$

$$\boldsymbol{s}_i \triangleq \text{col}\{\boldsymbol{s}_{1,i}, \ldots, \boldsymbol{s}_{N,i}\} \tag{16}$$

$$\mathcal{M} \triangleq \text{diag}\{\mu_1 I_M, \ldots, \mu_N I_M\} \tag{17}$$

$$\mathcal{G} \triangleq \text{diag}\{G, \ldots, G\} = I_N \otimes G \tag{18}$$

$$\mathcal{P} \triangleq \text{diag}\left\{P_G^\perp, \ldots, P_G^\perp\right\} = I_N \otimes P_G^\perp \tag{19}$$

$$\mathcal{H} \triangleq \text{diag}\{\Pi, \ldots, \Pi\} = I_N \otimes \Pi \tag{20}$$

$$\theta \triangleq \text{col}\{w^o, \ldots, w^o\} = \mathbb{1}_N \otimes w^o \tag{21}$$

$$\widetilde{\boldsymbol{w}}_i = \text{col}\{\widetilde{\boldsymbol{w}}_{1,i}, \ldots, \widetilde{\boldsymbol{w}}_{N,i}\} \tag{22}$$

Then, from (7), the recursion for the error vector $\widetilde{\boldsymbol{w}}_i$ can be found to be:

$$\boxed{\begin{aligned}\widetilde{\boldsymbol{w}}_i = \mathcal{A}^\mathsf{T}\mathcal{P}\left(I_{NM} - \mathcal{M}\mathcal{H} - \mathcal{M}\boldsymbol{\mathcal{R}}_i\right)\widetilde{\boldsymbol{w}}_{i-1} \\ -\mathcal{A}^\mathsf{T}\mathcal{P}\mathcal{M}\boldsymbol{s}_i + \mathcal{A}^\mathsf{T}\mathcal{P}\mathcal{M}\mathcal{H}\theta\end{aligned}} \tag{23}$$

where $\mathcal{A} = A \otimes I_M$.

4. MEAN-SQUARE CONVERGENCE ANALYSIS

Before we conduct the mean-square convergence analysis, we need to introduce the following assumptions on the statistical properties of the data.

Assumption 1 (Statistical properties of the data)**.**

1. *The regression data $\boldsymbol{u}_{k,i}$ are temporally white and spatially independent random variables with zero mean and covariance matrix $R_{u,k} \triangleq \mathbb{E}\boldsymbol{u}_{k,i}^* \boldsymbol{u}_{k,i} > 0$.*
2. *The noise signals $\boldsymbol{v}_k(i)$ are temporally white and spatially independent random variables with zero mean and variances $\sigma_{v,k}^2$.*
3. *The regression data $\boldsymbol{u}_{k,i}$ and noise signals $\boldsymbol{v}_l(j)$ are independent of each other for all k and l, i and j.* ■

It is worth noting that we do not assume Gaussian distributions for either the regressors or the noise signals.

4.1. Convergence in the Mean

From (8) and (12), we get

$$\begin{aligned} P_G^\perp \mathbb{E}\,\widetilde{\boldsymbol{w}}_{k,i} &= \mathbb{E} P_G^\perp \widetilde{\boldsymbol{w}}_{k,i} \\ &= \mathbb{E}\,[I - G(G^*G)^{-1}G^*]\widetilde{\boldsymbol{w}}_{k,i} \\ &= \mathbb{E}\,[\widetilde{\boldsymbol{w}}_{k,i} - G(G^*G)^{-1}G^*(w^o - \boldsymbol{w}_{k,i})] \\ &= \mathbb{E}\,\{\widetilde{\boldsymbol{w}}_{k,i} - [G(G^*G)^{-1}b - G(G^*G)^{-1}b]\} \\ &= \mathbb{E}\,\widetilde{\boldsymbol{w}}_{k,i} \end{aligned} \tag{24}$$

for $k = 1, 2, \ldots, N$. Then, from (19) and (22), we obtain

$$\mathcal{P}\mathbb{E}\widetilde{\boldsymbol{w}}_i = \mathbb{E}\widetilde{\boldsymbol{w}}_i \tag{25}$$

Using the fact that $(P_G^\perp)^2 = P_G^\perp$, or $\mathcal{P}^2 = \mathcal{P}$ by (19), we conclude from (23) that the network mean error vector satisfies the recursion:

$$\mathbb{E}\widetilde{\boldsymbol{w}}_i = \mathcal{B}\mathbb{E}\widetilde{\boldsymbol{w}}_{i-1} + \mathcal{A}^\mathsf{T}\mathcal{P}\mathcal{M}\mathcal{H}\theta \tag{26}$$

where

$$\mathcal{B} \triangleq \mathcal{A}^\mathsf{T}\mathcal{P}\left(I_{NM} - \mathcal{M}\mathcal{H} - \mathcal{M}\mathcal{R}_u\right)\mathcal{P} \tag{27}$$

$$\mathcal{R}_u \triangleq \mathbb{E}\boldsymbol{\mathcal{R}}_i \triangleq \text{diag}\{R_{u,1}, \ldots, R_{u,N}\} \tag{28}$$

Using the block maximum norm [15] with block size $M \times M$, it can be verified that

$$\begin{aligned} \rho(\mathcal{B}) &\le \|\mathcal{A}^\mathsf{T}\mathcal{P}(I_{NM} - \mathcal{M}\mathcal{H} - \mathcal{M}\mathcal{R}_u)\mathcal{P}\|_{b,\infty} \\ &\le \|\mathcal{A}^\mathsf{T}\|_{b,\infty} \cdot \|\mathcal{P}\|_{b,\infty} \cdot \|I_{NM} - \mathcal{M}\mathcal{H} - \mathcal{M}\mathcal{R}_u\|_{b,\infty} \cdot \|\mathcal{P}\|_{b,\infty} \\ &= \|\mathcal{P}\|_{b,\infty}^2 \cdot \rho(I_{NM} - \mathcal{M}\mathcal{H} - \mathcal{M}\mathcal{R}_u) \\ &= [\rho(P_G^\perp)]^2 \cdot \rho(I_{NM} - \mathcal{M}\mathcal{H} - \mathcal{M}\mathcal{R}_u) \\ &= \rho(I_{NM} - \mathcal{M}\mathcal{H} - \mathcal{M}\mathcal{R}_u) \end{aligned} \tag{29}$$

since $\|\mathcal{A}^\mathsf{T}\|_{b,\infty} = 1$ and $\rho(P_G^\perp) = 1$. From (29), we can get a sufficient condition for mean stability:

$$\boxed{\mu_k < \frac{2}{\rho\left(\Pi + R_{u,k}\right)}} \tag{30}$$

for $k = 1, 2, \ldots, N$. If condition (30) is satisfied, then recursion (26) will converge to

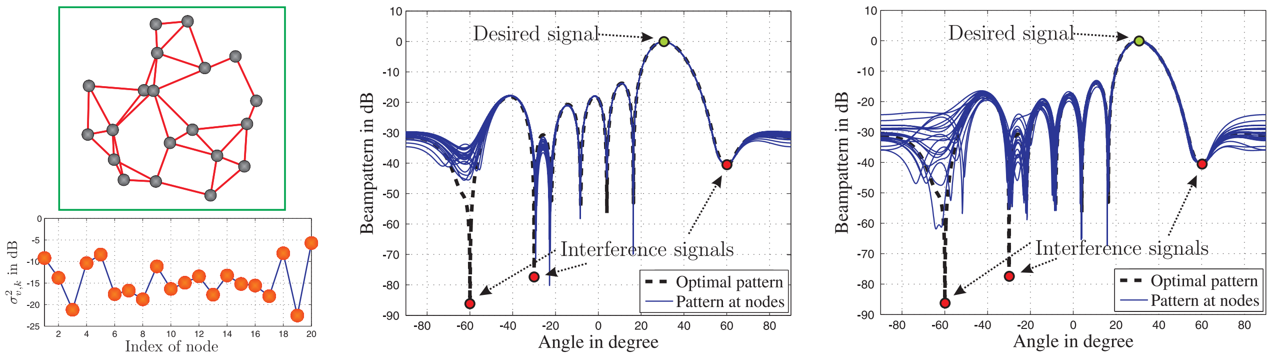
$$\lim_{i\to\infty} \mathbb{E}\widetilde{\boldsymbol{w}}_i = (I_{NM} - \mathcal{B})^{-1}\mathcal{A}^\mathsf{T}\mathcal{P}\mathcal{M}\mathcal{H}\theta \tag{31}$$

which is proportional to Π through $\mathcal{H}$ and is not dependent on the initial values $\{w_{k,-1}\}$. From (31) we know that if $\Pi \neq 0$, the steady-state estimator is biased. Nevertheless, if we choose the weighting matrix Π to be sufficiently small, say, $\Pi = \epsilon I_M$ where $0 < \epsilon \ll 1$, then the steady-state bias will be of the order of $O(\epsilon) \ll 1$; if we further allow Π to gradually diminish as the algorithm converges, then we will arrive at an unbiased estimator asymptotically.

4.2. Mean-Square Convergence

Starting from (23) we can show that

$$\boxed{\begin{aligned}\mathbb{E}\|\widetilde{\boldsymbol{w}}_i\|_\Sigma^2 = \mathbb{E}\|\widetilde{\boldsymbol{w}}_{i-1}\|_{\Sigma'}^2 + 2\text{Re}\left(\mathbb{E}\widetilde{\boldsymbol{w}}_{i-1}^*\mathcal{B}^*\Sigma\eta\right) \\ +\|\eta\|_\Sigma^2 + \mathbb{E}\|\mathcal{A}^\mathsf{T}\mathcal{P}\mathcal{M}\boldsymbol{s}_i\|_\Sigma^2\end{aligned}} \tag{32}$$



(a) Topology and noise profile (b) Beampattern of diffusion strategy. (c) Beampattern of non-cooperative strategy.

Fig. 2. Simulation setup and beampatterns resulting from diffusion cooperation and non-cooperation.

for any $NM \times NM$ positive semi-definite matrix Σ, where

$$\eta \triangleq \mathcal{A}^{\mathsf{T}} \mathcal{P} \mathcal{M} \mathcal{H} \theta \tag{33}$$

$$\Sigma' \triangleq \mathcal{B}^* \Sigma \mathcal{B} + O(\mathcal{M}^2) \tag{34}$$

The notation $O(\mathcal{M}^2)$ refers to a term of the order of $\mathcal{M}^2$. We further introduce a small step-size assumption so that terms that depend on higher-order powers of the step-sizes can be ignored.

Assumption 2 (Small step-sizes)**.** *The step-sizes are sufficiently small and satisfy condition* (30)*, i.e.,* $\mu_k \ll 1$. ■

Let us introduce the vector notation:

$$\sigma \triangleq \text{vec}(\Sigma) \tag{35}$$

$$\sigma' \triangleq \text{vec}(\Sigma') \approx \mathcal{F}\sigma \tag{36}$$

where

$$\mathcal{F} \triangleq \mathcal{B}^{\mathsf{T}} \otimes \mathcal{B}^* \tag{37}$$

Moreover, it can be verified that

$$\|\eta\|^2_{\Sigma} + \mathbb{E}\|\mathcal{A}^{\mathsf{T}}\mathcal{P}\mathcal{M}\boldsymbol{s}_i\|^2_{\Sigma} = [\text{vec}(\mathcal{Y})]^* \sigma \tag{38}$$

$$2\text{Re}\left(\mathbb{E}\widetilde{\boldsymbol{w}}^*_{i-1}\mathcal{B}^*\Sigma\eta\right) = [\text{vec}(\mathcal{Z}_{i-1})]^*\sigma \tag{39}$$

where

$$\mathcal{Y} \triangleq \mathcal{A}^{\mathsf{T}}\mathcal{P}\mathcal{M}\mathcal{S}\mathcal{M}\mathcal{P}\mathcal{A} + \eta\eta^* \tag{40}$$

$$\mathcal{Z}_{i-1} \triangleq \eta(\mathbb{E}\widetilde{\boldsymbol{w}}^*_{i-1})\mathcal{B}^* + \mathcal{B}(\mathbb{E}\widetilde{\boldsymbol{w}}_{i-1})\eta^* \tag{41}$$

$$\mathcal{S} \triangleq \mathbb{E}\boldsymbol{s}_i\boldsymbol{s}_i^* = \text{diag}\left\{\sigma^2_{v,1}R_{u,1}, \ldots, \sigma^2_{v,N}R_{u,N}\right\} \tag{42}$$

Then, expression (32) can be approximately rewritten as

$$\boxed{\mathbb{E}\|\widetilde{\boldsymbol{w}}_i\|^2_{\sigma} \approx \mathbb{E}\|\widetilde{\boldsymbol{w}}_{i-1}\|^2_{\mathcal{F}\sigma} + [\text{vec}(\mathcal{Y} + \mathcal{Z}_{i-1})]^*\sigma} \tag{43}$$

Under Assumption 2, the mean error recursion (26) converges. Therefore, the sequences $\{\mathbb{E}\widetilde{\boldsymbol{w}}_i, \mathcal{Z}_i\}$ can be uniformly bounded. Then, the weighted variance relation (43) converges if the matrix $\mathcal{F}$ is stable, i.e., $\rho(\mathcal{F}) < 1$, which is equivalent to $\rho(\mathcal{B}) < 1$ according to (37). This condition is the same as the condition for mean stability.

Theorem 1 (Mean-square convergence condition)**.** *The diffusion strategy* (7) *is stable in the mean and mean-square sense if the step-sizes are sufficiently small and satisfy* (30)*.* ■

5. STEADY-STATE PERFORMANCE ANALYSIS

At steady-state as $i \to \infty$, from (43), we can get the steady-state relation:

$$\lim_{i\to\infty} \mathbb{E}\|\widetilde{\boldsymbol{w}}_i\|^2_{(I_{N^2M^2}-\mathcal{F})\sigma} \approx [\text{vec}(\mathcal{Y} + \mathcal{Z}_\infty)]^*\sigma \tag{44}$$

where, by (31), (33), and (41),

$$\mathcal{Z}_\infty \triangleq \lim_{i\to\infty} \mathcal{Z}_i = \eta\widetilde{w}^*_\infty\mathcal{B}^* + \mathcal{B}\widetilde{w}_\infty\eta^* \tag{45}$$

$$\widetilde{w}_\infty \triangleq \lim_{i\to\infty} \mathbb{E}\widetilde{\boldsymbol{w}}_i = (I_{NM} - \mathcal{B})^{-1}\eta \tag{46}$$

The network MSD is defined as

$$\overline{\text{MSD}} \triangleq \lim_{i\to\infty} \frac{1}{N}\sum_{k=1}^{N} \mathbb{E}\|\widetilde{\boldsymbol{w}}_{k,i}\|^2 \tag{47}$$

Selecting σ as $(I_{N^2M^2} - \mathcal{F})\sigma = \text{vec}(I_{NM}/N)$, expression (44) gives

$$\boxed{\overline{\text{MSD}} \approx \frac{1}{N}[\text{vec}(\mathcal{Y} + \mathcal{Z}_\infty)]^*(I_{N^2M^2} - \mathcal{F})^{-1}\text{vec}(I_{NM})} \tag{48}$$

Similarly, if we instead select $(I_{N^2M^2} - \mathcal{F})\sigma = \text{vec}(\mathcal{R}_u/N)$, then expression (44) allows us to evaluate the network EMSE as:

$$\boxed{\overline{\text{EMSE}} \approx \frac{1}{N}[\text{vec}(\mathcal{Y} + \mathcal{Z}_\infty)]^*(I_{N^2M^2} - \mathcal{F})^{-1}\text{vec}(\mathcal{R}_u)} \tag{49}$$

where the network EMSE is defined as follows:

$$\overline{\text{EMSE}} \triangleq \lim_{i\to\infty} \frac{1}{N}\sum_{k=1}^{N} \mathbb{E}|\boldsymbol{u}_{k,i}\widetilde{\boldsymbol{w}}_{k,i-1}|^2 \tag{50}$$

6. SIMULATION RESULTS AND CONCLUSIONS

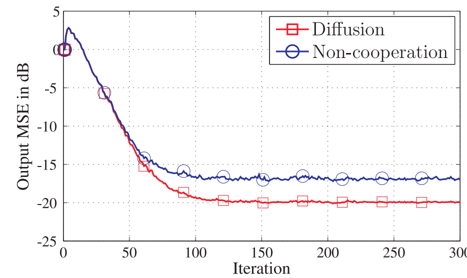
We simulate the diffusion algorithm versus non-cooperative stand-alone LMS updates at each array over the connected network with $N = 20$ nodes shown in the upper part of Fig. 2a, where each node represents an array. Each array has $M = 10$ elements. The noise variance profile is shown in the lower part of Fig. 2a. The DOA and power of the desired and interference signals are given in Table 2. We assume that the DOA of $\boldsymbol{s}_3(i)$ is roughly known so that two affine constraints are introduced to enforce the responses at 58.5° and 61.5° to be -40 dB.

The resulting beampattern at each node is plotted in blue curves in Fig. 2b for the diffusion strategy and in Fig. 2c for non-cooperative LMS. The dashed black curve is the beampattern obtained from solving problem (5) with full knowledge of data statistics. We observe that, the beampatterns generated by the diffusion strategy are more concentrated around the optimal beampattern, and have lower sidelobes. The performance is more uniform across the network; there are less fluctuations in the sidelobe area ($5 \sim 10$ dB) for diffusion compared to the non-cooperative LMS case (about $10 \sim 35$ dB) across the nodes. In Fig. 3a–3b, we show the learning curves of network MSE and network MSD for both diffusion and non-cooperative stand-alone LMS strategies, which are averaged over 1000 trials. The network MSE (MSD) is obtained by averaging over all arrays across the network. We note that the diffusion algorithm can improve the steady-state performance by about 3.5 dB.

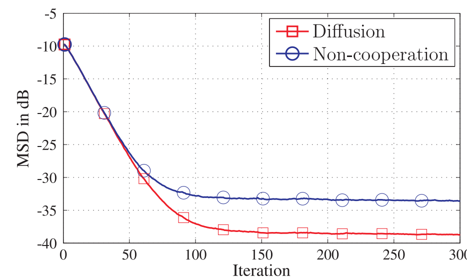
Table 2. Simulation Parameters.

	$s_0(i)$	$s_1(i)$	$s_2(i)$	$s_3(i)$
DOA (degree)	30	-60	-30	60
Power (dB)	0	12.4	5.5	6.3

We examine the theoretical result (48) in Fig. 4. We maintain the network topology and noise profile from Fig. 2a and use the same parameters from Table 2. We reduce the number of antenna elements for the arrays by setting $M = 4$ to reduce the simulation time. Fig. 4 indicates that the theoretical MSD expression matches well with simulation.



(a) MSE learning curves.



(b) MSD learning curves.

Fig. 3. Comparison of learning curves.

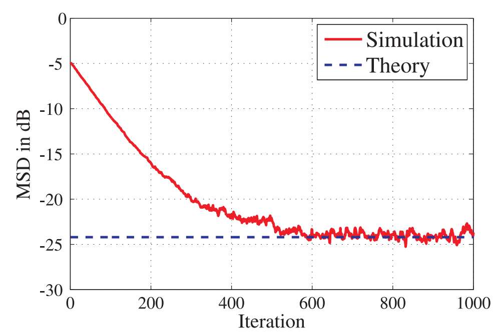


Fig. 4. MSD learning curve for the diffusion strategy when $M = 4$.

7. REFERENCES

[1] M. Wax and T. Kailath, "Decentralized processing in sensor arrays," *IEEE Trans. Signal Process.*, vol. 33, no. 4, pp. 1123–1129, Oct. 1985.

[2] G. Barriac, R. Mudumbai, and U. Madhow, "Distributed beamforming for information transfer in sensor netowrks," in *Proc. ACM/IEEE Int. Conf. Inform. Process. Sensor Networks (IPSN)*, Berkeley, CA, Apr. 2004, pp. 81–88.

[3] H. Ochiai, P. Mitran, H. V. Poor, and V. Tarokh, "Collaborative beamforming for distributed wireless Ad Hoc sensor netowrks," *IEEE Trans. Signal Process.*, vol. 53, no. 11, pp. 4110–4124, Nov. 2005.

[4] M. F. A. Ahmed and S. A. Vorobyov, "Collaborative beamforming for wireless sensor netowrks with Gaussian distributed sensor nodes," *IEEE Trans. Wireless Commun.*, vol. 8, no. 2, pp. 638–643, Feb. 2009.

[5] X. Wang, H. Ge, and I. P. Kirsteins, "Direction-of-arrival estimation using distributed arrays: A canonical coordinates perspective with limited array size and sample support," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process. (ICASSP)*, Dallas, TX, Mar. 2010, pp. 2622–2625.

[6] H. Ge, I. P. Kirsteins, and X. Wang, "Adaptive beamforming using distributed antenna arrays: Joint versus distributed processing," in *Proc. Asilomar Conf. Signals, Systems, Computers*, Pacific Grove, CA, Nov. 2010, pp. 1107–1111.

[7] C. G. Lopes and A. H. Sayed, "Diffusion least-mean squares over adaptive networks: Formulation and performance analysis," *IEEE Trans. Signal Process.*, vol. 56, no. 7, pp. 3122–3136, July 2008.

[8] F. S. Cattivelli and A. H. Sayed, "Diffusion LMS strategies for distributed estimation," *IEEE Trans. Signal Process.*, vol. 58, no. 3, pp. 1035–1048, Mar. 2010.

[9] S-Y. Tu and A. H. Sayed, "Mobile adaptive networks," *IEEE J. Sel. Top. Signal Process.*, vol. 5, no. 4, pp. 649–664, Aug. 2011.

[10] F. S. Cattivelli and A. H. Sayed, "Modeling bird flight formations using diffusion adaptation," *IEEE Trans. Signal Process.*, vol. 59, no. 5, pp. 2038–2051, May 2011.

[11] J. Chen and A. H. Sayed, "Diffusion adaptation strategies for distributed optimization and learning over networks," arXiv:1111.0034v1 [math.OC], Oct. 2011.

[12] H. L. Van Trees, *Optimum Array Processing*, Wiley, NJ, 2002.

[13] X. Mestre and M. A. Lagunas, "Diagonal loading for finite sample size beamforming: an asymptotic approach," in *Robust Adaptive Beamforming*, J. Li and P. Stoica, Eds., pp. 201–257. Wiley, NJ, 2005.

[14] S. Theodoridis, K. Slavakis, and I. Yamada, "Adaptive learning in a world of projections: A unifying framework for linear and nonlinear classification and regression tasks," *IEEE Signal Process. Mag.*, vol. 28, no. 1, pp. 97–123, Jan. 2011.

[15] N. Takahashi, I. Yamada, and A. H. Sayed, "Diffusion least-mean squares with adaptive combiners: Formulation and performance analysis," *IEEE Trans. Signal Process.*, vol. 58, no. 9, pp. 4795–4810, Sept. 2010.