

SEQUENTIAL DIRECTION DETECTION FOR SOUND SCENE ANALYSIS

Nail A. Gumerov,* Bowen Zhi,* and Ramani Duraiswami*

VisiSonics Corporation, College Park, MD 20742

ABSTRACT

We introduce a novel algorithm for the decomposition of a broadband soundfield into its component plane-waves. The algorithm, termed **Sequential Direction Detection**, decomposes the soundfield into L plane waves by recursively minimizing an objective function that determines the plane-wave directions, strengths and the number of plane-waves. The algorithm is described and tested on synthetic and real data. Extensions are discussed.

Index Terms— Array Processing, Planewave decomposition

1. INTRODUCTION

The soundfield at a point in any environment carries a tremendous amount of information, which is used by a listener to understand source locations, message content, and the size and ambience of the space. It would be useful to decompose the sound into its components for identification, and obtain the location/direction and content of individual source objects, especially in applications recreating real scenes in virtual and augmented reality, where sources are usually broadband. Microphone arrays are often used for this. An issue faced is the lack of algorithms to perform such decompositions reliably. As such, steered beamforming has often been used [3, 2]. Plane-wave decomposition with arrays of special shape, such as spherical/cylindrical [5, 7], has been suggested. However in these cases the number of sources and their directions are not estimated.

Here, we look at the problem of incident field reconstruction at a location by imposing the prior that the scene is generated by an unknown number of distant broadband sources, which is collected at a spatially compact microphone array of M microphones. The signal from these sources (or their reflections) arrive at the array and can be modeled as far-field plane-waves incident from various directions. Imposing this prior, we develop a formulation for identifying the incoming plane-wave directions via computing a cost function based on those frequencies for which the array theoretically exhibits no aliasing. We employ a novel sequential operator formulation, which identifies successively the leading order plane-waves. After identifying the directions, we are able to build a plane-wave representation over the entire audible frequency range for these directions. Results from synthetic experiments are presented, along with a real demonstration.

2. PROBLEM STATEMENT

We consider a broadband acoustic field received at an array of M sensors (microphones). The field is assumed to be created by an unknown number of plane-waves L . After converting a frame of data to the frequency domain, assume that there are N frequencies, and the field at each frequency at a point \mathbf{r} is

$$p_n(\mathbf{r}) = \sum_{l=1}^L A_{nl} e^{-ik_n \mathbf{s}_l \cdot \mathbf{r}}, \quad k_n = \frac{\omega_n}{C}, \quad n = 1, \dots, N, \quad (1)$$

where \mathbf{s}_l are the directions of arrival (DOA), ω_n are the circular frequencies with wave-numbers k_n , and A_{nl} the complex amplitudes. For microphone locations $\mathbf{r}_1, \dots, \mathbf{r}_M$, the system of equations describing microphone readings can be written in the form

$$\sum_{l=1}^L A_{nl} e^{-ik_n \mathbf{s}_l \cdot \mathbf{r}_m} = p_n(\mathbf{r}_m), \quad m = 1, \dots, M, \quad n = 1, \dots, N, \quad (2)$$

or in matrix-vector form

$$\mathbf{H}_n \mathbf{A}_n = \mathbf{P}_n, \quad n = 1, \dots, N, \quad (3)$$

where \mathbf{H}_n is a $M \times L$ matrix with entries $(\mathbf{H}_n)_{ml} = e^{-ik_n \mathbf{s}_l \cdot \mathbf{r}_m}$, \mathbf{A}_n a L vector with entries $(\mathbf{A}_n)_l = A_{nl}$, and \mathbf{P}_n the M vector with entries $(\mathbf{P}_n)_m = p_n(\mathbf{r}_m)$. Then

$$\mathbf{H}_n = (\mathbf{h}_n(\mathbf{s}_1) \quad \mathbf{h}_n(\mathbf{s}_2) \quad \dots \quad \mathbf{h}_n(\mathbf{s}_L)), \quad (4)$$

where $\mathbf{h}_n(\mathbf{s}_l)$ are M vectors, known as “steering” vectors, while \mathbf{H}_n is called the “steering matrix”.

The problem is: given \mathbf{P}_n , determine L , the DOA $\mathbf{s}_1, \dots, \mathbf{s}_L$, and amplitudes $\{A_{nl}\}$. The field in (1) is characterized by NL complex amplitudes A_{nl} and L unit vectors \mathbf{s}_l , or $2(N+1)L$ real unknowns for 3D (two angles/direction) and $(2N+1)L$ unknowns in 2D (one angle/direction). We assume directions are consistent across frequencies (i.e., sources are broadband). The microphone readings provide NM complex numbers p_{mn} which yield $2NM$ equations using (2) and (3). The system can be solved if

$$L \leq \frac{MN}{N+1} < M, \text{ in } \mathbb{R}^3, \quad L \leq \frac{2MN}{2N+1} < M, \text{ in } \mathbb{R}^2. \quad (5)$$

This shows that as the number of frequencies N (or bandwidth) increases, the number of detectable DOA also increases. Regardless, L must be smaller than the number of microphones M .

3. SEQUENTIAL DIRECTION DETECTION ALGORITHM

The solution of (3) can be sought by globally minimizing a suitable cost function based on discrepancy between measured and predicted data with respect to $\{\mathbf{A}_n\}$ and $\{\mathbf{s}_l\}$, in a suitable norm such as L_2 ,

$$\mathcal{F} = \sum_{n=1}^N w_n \|\mathbf{H}_n \mathbf{A}_n - \mathbf{P}_n\|_2^2 \rightarrow \min, \quad (6)$$

where w_n are some positive weights (e.g. $w_n = 1, n = 1, \dots, N$).

Note that $\{\mathbf{s}_l\}$ determine only \mathbf{H}_n . Hence, the minimum of the functional (6) should be achieved when the amplitudes \mathbf{A}_n are related to \mathbf{P}_n via minimization for a given \mathbf{H}_n , which in L_2 is

$$\mathbf{A}_n = (\mathbf{H}_n^* \mathbf{H}_n)^{-1} \mathbf{H}_n^* \mathbf{P}_n, \quad n = 1, \dots, N, \quad (7)$$

where \mathbf{H}_n^* is the transpose conjugate of \mathbf{H}_n and we assume $\mathbf{H}_n^* \mathbf{H}_n$ is pseudo-invertible. On the other hand, this relation determines the

* Also at University of Maryland, College Park.

optimal \mathbf{A}_n as functions of directions $\{\mathbf{s}_l\}$. Substituting Eq. (7) into Eq. (6), we can see that the number of independent variables for the objective function reduces to L directions \mathbf{s}_l , as we have

$$\mathcal{F}(\mathbf{s}_1, \dots, \mathbf{s}_L) = \sum_{n=1}^N w_n \mathcal{F}_n, \quad (8)$$

$$\mathcal{F}_n = \left\| \left(\mathbf{H}_n (\mathbf{H}_n^* \mathbf{H}_n)^{-1} \mathbf{H}_n^* - \mathbf{I} \right) \mathbf{P}_n \right\|_2^2,$$

where \mathbf{I} is the $L \times L$ identity matrix.

Despite the reduction in dimension at this stage by only considering directions (from $2(N+1)L$ to $2L$ in 3D, and $(2N+1)L$ to L in 2D), nonlinear optimization is still expensive in L (unknown) dimensions. Further, multiple local minima complicate the search for the global minimum. We propose here a method for approximate determination of the directions, which has relatively low computational complexity. We assume

$$\mathbf{s} \neq \mathbf{t}, \Rightarrow \mathbf{h}_n(\mathbf{s}) \neq \mathbf{h}_n(\mathbf{t}), \quad n = 1, \dots, N. \quad (9)$$

SDD constructs steering matrices \mathbf{H}_n via consequent determination of optimal directions $\mathbf{s}_1, \mathbf{s}_2, \dots$ terminated by an exit criteria. At the l th step the $M \times l$ steering matrix, which is a function of \mathbf{s} , is

$$\mathbf{H}_n^{(l)}(\mathbf{s}) = \begin{pmatrix} \mathbf{h}_n^{(1)} & \dots & \mathbf{h}_n^{(l-1)} & \mathbf{h}_n(\mathbf{s}) \end{pmatrix}. \quad (10)$$

Here $\mathbf{h}_n^{(k)} = \mathbf{h}_n(\mathbf{s}_k)$, $k = 1, \dots, l-1$, are constants, as the directions $\mathbf{s}_1, \dots, \mathbf{s}_{l-1}$ are determined at earlier steps. We consider then the objective function $\mathcal{F}^{(l)}(\mathbf{s})$,

$$\mathcal{F}^{(l)}(\mathbf{s}) = \sum_{n=1}^N w_n \mathcal{F}_n^{(l)}(\mathbf{s}), \quad (11)$$

$$\mathcal{F}_n^{(l)}(\mathbf{s}) = \left\| \left(\mathbf{H}_n^{(l)} \left(\mathbf{H}_n^{(l)*} \mathbf{H}_n^{(l)} \right)^{-1} \mathbf{H}_n^{(l)*} - \mathbf{I} \right) \mathbf{P}_n \right\|_2^2,$$

which is globally minimized at $\mathbf{s} = \mathbf{s}_l$ and continue recursively, assigning $\mathbf{h}_n^{(l)} = \mathbf{h}_n(\mathbf{s}_l)$ and setting the steering matrix $\mathbf{H}_n^{(l)}(\mathbf{s}_l)$ at the l th iteration to \mathbf{H}_n . The iteration terminates at $l = M-1$ or

$$\epsilon^{(l)} = \left(\frac{\mathcal{F}^{(l)}(\mathbf{s}_l)}{\sum_{n=1}^N w_n \|\mathbf{P}_n\|_2^2} \right)^{1/2} < \epsilon_{tol}, \quad (12)$$

where ϵ_{tol} is the tolerance and $\epsilon^{(l)}$ is the relative error in the L_2 norm, for $\mathbf{H}_n = \mathbf{H}_n^{(l)}(\mathbf{s}_l)$. Consider now the first step of the algorithm at which we should determine \mathbf{s}_1 . This corresponds to the guess that the field is generated by one plane wave. Then $\mathbf{H}_n^{(1)}(\mathbf{s})$ has size $M \times 1$ and consists of one vector $\mathbf{h}_n(\mathbf{s})$. The objective function for the first step is,

$$\mathcal{F}^{(1)}(\mathbf{s}) = \sum_{n=1}^N w_n \mathcal{F}_n^{(1)}(\mathbf{s}), \quad \mathcal{F}_n^{(1)}(\mathbf{s}) = \mathbf{P}_n^* \left(\mathbf{I} - \frac{\mathbf{h}_n(\mathbf{s}) \mathbf{h}_n^*(\mathbf{s})}{\mathbf{h}_n^*(\mathbf{s}) \mathbf{h}_n(\mathbf{s})} \right) \mathbf{P}_n. \quad (13)$$

The global minimum of any $\mathcal{F}^{(l)}(\mathbf{s})$ over the two angles (in 3D) or one angle (in 2D) is relatively easily found, (e.g., using gradient methods). Denote the minimum as \mathbf{s}_l and check if the incident field is well approximated by l plane-waves using Eq. (12). If $l < M-1$ and condition (12) does not hold we go to the $l+1$ th step.

3.1. Recursion for SDD operators

The computational complexity of the implementation using Eqs (10) and (11) directly increases with l . This is due to several matrix-matrix multiplications and matrix inversion operations, which cost $O(l^3) + O(M^2)$ for the l th step. This can be reduced to $O(M^2)$ using a recursive process for generating the SDD operators, namely $M \times M$ matrices $\mathbf{L}_n^{(l)}(\mathbf{s})$,

$$\mathbf{L}_n^{(l)}(\mathbf{s}) = \mathbf{I} - \mathbf{H}_n^{(l)}(\mathbf{s}) \mathbf{G}_n^{(l)}(\mathbf{s}) \mathbf{H}_n^{(l)*}(\mathbf{s}), \quad (14)$$

where

$$\mathbf{G}_n^{(l)}(\mathbf{s}) = \left(\mathbf{H}_n^{(l)*}(\mathbf{s}) \mathbf{H}_n^{(l)}(\mathbf{s}) \right)^{-1}. \quad (15)$$

The objective function for the l th step takes the form

$$\mathcal{F}^{(l)}(\mathbf{s}) = \sum_{n=1}^N w_n \left\| \mathbf{L}_n^{(l)}(\mathbf{s}) \mathbf{P}_n \right\|_2^2 = \sum_{n=1}^N w_n \mathbf{P}_n^* \mathbf{L}_n^{(l)}(\mathbf{s}) \mathbf{P}_n. \quad (16)$$

For constant matrices, computed at step $l-1$ we will use notation $\mathbf{L}_n^{(l-1)} = \mathbf{L}_n^{(l-1)}(\mathbf{s}_{l-1})$, $\mathbf{G}_n^{(l-1)} = \mathbf{G}_n^{(l-1)}(\mathbf{s}_{l-1})$, and $\mathbf{H}_n^{(l-1)} = \mathbf{H}_n^{(l-1)}(\mathbf{s}_{l-1})$. Also, for brevity we drop argument \mathbf{s} of matrix functions $\mathbf{L}_n^{(l)}$, $\mathbf{G}_n^{(l)}$, $\mathbf{H}_n^{(l)}$, and vector function \mathbf{h}_n . Representing

$$\mathbf{H}_n^{(l)} = \begin{pmatrix} \mathbf{H}_n^{(l-1)} & \mathbf{h}_n \end{pmatrix}, \quad (17)$$

we obtain

$$\mathbf{G}_n^{(l)} = \left(\begin{pmatrix} \mathbf{G}_n^{(l-1)} & \mathbf{H}_n^{(l-1)*} \mathbf{h}_n \\ \mathbf{h}_n^* \mathbf{H}_n^{(l-1)} & \mathbf{h}_n^* \mathbf{h}_n \end{pmatrix}^{-1} \right), \quad (18)$$

and $\left(\mathbf{G}_n^{(l-1)} \right)^{-1} = \mathbf{H}_n^{(l-1)*} \mathbf{H}_n^{(l-1)}$. We use the following formula for an arbitrary (invertible) block matrix,

$$G = \begin{pmatrix} A & B \\ C & D \end{pmatrix}^{-1} = \begin{pmatrix} A^{-1} + A^{-1} B E C A^{-1} & -A^{-1} B E \\ -E C A^{-1} & E \end{pmatrix} \quad (19)$$

with $E = (D - C A^{-1} B)^{-1}$. When D is scalar, E is also one, so

$$G = \begin{pmatrix} A^{-1} & 0 \\ 0 & 0 \end{pmatrix} + E \begin{pmatrix} A^{-1} B C A^{-1} & -A^{-1} B \\ -C A^{-1} & 1 \end{pmatrix}. \quad (20)$$

In our case we should set

$$\begin{aligned} G &= \mathbf{G}_n^{(l)}, \quad A^{-1} = \mathbf{G}_n^{(l-1)}, \quad B = \mathbf{H}_n^{(l-1)*} \mathbf{h}_n, \quad C = \mathbf{h}_n^* \mathbf{H}_n^{(l-1)} = B^*, \\ E^{-1} &= \mathbf{h}_n^* \mathbf{h}_n - \mathbf{h}_n^* \mathbf{H}_n^{(l-1)} \mathbf{G}_n^{(l-1)} \mathbf{H}_n^{(l-1)*} \mathbf{h}_n = \mathbf{h}_n^* \mathbf{L}_n^{(l-1)} \mathbf{h}_n. \end{aligned} \quad (21)$$

Substituting this into definition (14) and simplifying,

$$\mathbf{L}_n^{(l)} = \mathbf{L}_n^{(l-1)} - \frac{\mathbf{L}_n^{(l-1)} \mathbf{h}_n \mathbf{h}_n^* \mathbf{L}_n^{(l-1)}}{\mathbf{h}_n^* \mathbf{L}_n^{(l-1)} \mathbf{h}_n}, \quad l = 1, 2, \dots \quad (22)$$

For $l = 1$, we set $\mathbf{L}_n^{(0)} = \mathbf{I}$. Eq. (22) has stored constant matrices $\mathbf{L}_n^{(l-1)}$ to compute $\mathcal{F}^{(l)}(\mathbf{s})$ (see Eq. (16)), which thus requires only a few M matrix-vector multiplications. As soon as the optimal direction \mathbf{s}_l is found, the constant matrix $\mathbf{L}_n^{(l)}(\mathbf{s}_l)$ needed for the $l+1$ th iteration can be computed using Eq. (22), also taking $O(M^2)$ operations. The total complexity of the recursive algorithm for the maximum number of steps is $O(M^3)$ as opposed to $O(M^4)$.

Equation (22) reveals a number of features about the SDD algorithm. *First*, for any \mathbf{s} , the steering vector $\mathbf{h}_n(\mathbf{s})$ is an eigenvector

of $\mathbf{L}_n^{(l)}(\mathbf{s})$ corresponding to zero eigenvalue, or belongs to the nullspace of $\mathbf{L}_n^{(l)}(\mathbf{s})$. Indeed, as immediately follows from Eq. (22)

$$\mathbf{L}_n^{(l)} \mathbf{h}_n = \mathbf{L}_n^{(l-1)} \mathbf{h}_n - \frac{\mathbf{L}_n^{(l-1)} \mathbf{h}_n \mathbf{h}_n^* \mathbf{L}_n^{(l-1)} \mathbf{h}_n}{\mathbf{h}_n^* \mathbf{L}_n^{(l-1)} \mathbf{h}_n} \mathbf{h}_n = \mathbf{0}. \quad (23)$$

Second, Eq. (22) shows that any eigenvector of $\mathbf{L}_n^{(l-1)}$, $l > 1$, corresponding to zero eigenvalue will be also eigenvector of $\mathbf{L}_n^{(l)}$, so the nullspace of operator $\mathbf{L}_n^{(l)}$ includes the nullspace of operator $\mathbf{L}_n^{(l-1)}$. Therefore, we have by induction that all vectors $\mathbf{h}_n^{(1)}, \mathbf{h}_n^{(2)}, \dots, \mathbf{h}_n^{(l-1)}$ are the eigenvectors of $\mathbf{L}_n^{(l)}$ corresponding to zero eigenvalues.

Third, this shows that for $\mathbf{s} = \mathbf{s}_{l-1}$ we have $\mathbf{L}_n^{(l)}(\mathbf{s}) = \mathbf{L}_n^{(l-1)}$ and so $\mathcal{F}^{(l)}(\mathbf{s}) = \mathcal{F}^{(l-1)}(\mathbf{s}_{l-1})$. Therefore, $\min \mathcal{F}^{(l)}(\mathbf{s}) \leq \min \mathcal{F}^{(l-1)}(\mathbf{s}) = \mathcal{F}^{(l-1)}(\mathbf{s}_{l-1})$ and we have by induction

$$\min \mathcal{F}^{(l)} \leq \dots \leq \min \mathcal{F}^{(1)} \leq \mathcal{F}^{(0)} \equiv \sum_{n=1}^N w_n \|\mathbf{P}_n\|_2^2. \quad (24)$$

For success, we should have strict inequalities in Eq. (24). In this case the minimal $\mathcal{F}^{(l)}(\mathbf{s})$ should be at some $\mathbf{s} = \mathbf{s}_l \neq \mathbf{s}_{l-1}$. This also means that all directions found would be distinct.

Fourth, if we do have $\mathbf{s}_1, \dots, \mathbf{s}_l$ all different, the steering vectors $\mathbf{h}_n^{(1)}, \dots, \mathbf{h}_n^{(l)}$ corresponding to these directions are also different (see Eq. (9)). This means that in this case $\text{rank}(\mathbf{L}_n^{(l)}(\mathbf{s}_l)) = M - l$ since the nullspace of $\mathbf{L}_n^{(l)}(\mathbf{s}_l)$ is

$$\ker(\mathbf{L}_n^{(l)}) = \text{span}(\mathbf{h}_n^{(1)}, \dots, \mathbf{h}_n^{(l)}), \quad \dim(\ker(\mathbf{L}_n^{(l)})) = l. \quad (25)$$

This shows that $\mathbf{L}_n^{(M)}(\mathbf{s}) \equiv \mathbf{0}$, $\mathcal{F}^{(l)}(\mathbf{s}) \equiv 0$ for any \mathbf{s} , consistent with the fact that the maximum number of steps is $l = M - 1$.

3.2. SDD algorithm summary

We define the following $M \times M$ matrices $\mathbf{L}_n^{(l)}(\mathbf{s})$ and $M \times 1$ vectors $\mathbf{l}_n^{(l)}(\mathbf{s})$ as functions of direction \mathbf{s} :

$$\begin{aligned} \mathbf{L}_n^{(0)}(\mathbf{s}) &\equiv \mathbf{I}, \\ \mathbf{l}_n^{(l)}(\mathbf{s}) &= \mathbf{L}_n^{(l)}(\mathbf{s}_l) \mathbf{h}_n(\mathbf{s}), \quad l = 0, 1, 2, \dots, \\ \mathbf{L}_n^{(l)}(\mathbf{s}) &= \mathbf{L}_n^{(l-1)}(\mathbf{s}_{l-1}) - \frac{\mathbf{l}_n^{(l-1)}(\mathbf{s}) \mathbf{l}_n^{(l-1)*}(\mathbf{s})}{\mathbf{l}_n^{(l-1)*}(\mathbf{s}) \mathbf{l}_n^{(l-1)}(\mathbf{s})}, \quad l = 1, 2, \dots, \end{aligned} \quad (26)$$

where \mathbf{I} is the identity. We define the objective (steering) function as

$$\mathcal{F}_n^{(l)}(\mathbf{s}) = \mathbf{P}_n^* \mathbf{L}_n^{(l)}(\mathbf{s}) \mathbf{P}_n, \quad \mathcal{F}^{(l)}(\mathbf{s}) = \sum_{n=1}^N w_n \mathcal{F}_n^{(l)}(\mathbf{s}), \quad (27)$$

and the relative norm of the residual

$$\epsilon^{(l)} = \left(\frac{\mathcal{F}^{(l)}(\mathbf{s}_l)}{\|\mathbf{P}\|_2^2} \right)^{1/2}, \quad \|\mathbf{P}\|_2^2 = \sum_{n=1}^N w_n \|\mathbf{P}_n\|_2^2, \quad \|\mathbf{P}_n\|_2^2 = \mathbf{P}_n^* \mathbf{P}_n. \quad (28)$$

The SDD algorithm then is the following:

- Set some tolerance, $\epsilon_{tol} < 1$,
- compute and store $\|\mathbf{P}\|_2^2$
- set $l = 0$, $\epsilon^{(l)} = 1$, $\mathbf{L}_n^{(l)}(\mathbf{s}_l) = \mathbf{I}$.
- while $\epsilon^{(l)} > \epsilon_{tol}$
 1. $l = l + 1$;
 2. find and store $\mathbf{s}_l = \arg \min \mathcal{F}_{SDD}^{(l)}(\mathbf{s})$;
 3. evaluate $\mathbf{L}_n^{(l)}(\mathbf{s}_l)$;
 4. evaluate $\epsilon^{(l)}$;
- $L = l$; the required set of directions is $\{\mathbf{s}_1, \dots, \mathbf{s}_l\}$.

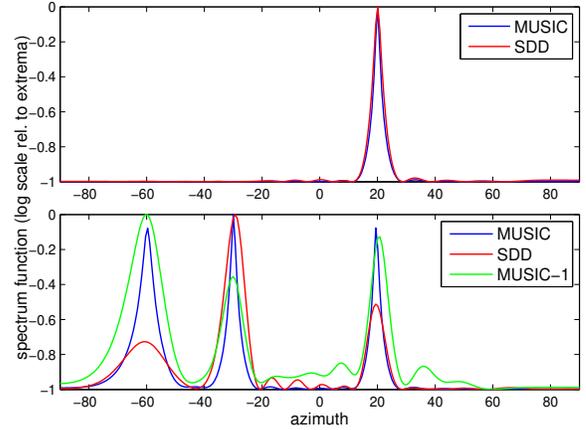


Fig. 1. The initial spectrum functions for experiments A (top) and B (bottom) plotted on a logarithmic scale, where MUSIC-1 refers to MUSIC with single-frame covariance estimation.

4. EXPERIMENTAL DEMONSTRATION

We perform a set of experiments based on simulated and real data. In each simulation we place a number of sources in a virtual room. Only direct paths are considered in simulations. Each source signal was independently generated pink noise. The simulated microphones are omnidirectional and record at 44.1 kHz. Gaussian white noise is added to each simulated recording with SNR of 10 dB.

We design a total of four synthetic experiments, labeled as A, B, C, D, and one real experiment labeled E. In the synthetic experiments, we compare the spectrum functions of MUSIC and SDD, where the SDD spectrum function is the reciprocal of its objective function (though note the SDD is performed over multiple iterations and uses a more complex cost function than what is plotted). In the real experiment, we compare the directions computed by SDD to ground truth. The frame size was selected to be 2048. We denote the azimuth and elevation of a source relative to the array center as (ϕ, θ) . Note that MUSIC is given the number of sources present and uses 4 frames of data to perform the modified covariance estimation described in [4], while SDD determines the number of sources and uses only one frame.

Experiment A: A horizontal 16-element uniform linear array with element spacing of 0.1 meters records a single source at $(20, 0)$. The algorithms process a single frequency band corresponding to ~ 1.5 kHz (the wavelength is roughly twice array spacing), and evaluate their spectrum at 256 equally-spaced points corresponding to azimuths between -90 and 90 degrees for both display and peak searching.

Experiment B: The configuration of this experiment is the same as that of experiment A, except the acoustic scene now consists of three sources located at $(20, 0)$, $(-30, 0)$, and $(-60, 0)$. An additional result was obtained with MUSIC using only a single frame for covariance estimation.

Experiment C: Extending the previous experiments in both azimuth and elevation, we simulate a recording using a 64-element array with microphones arranged in an equally-spaced 8×8 grid with spacing of 0.02 meters. The scene consists of four sources at $(20, -10)$, $(10, 25)$, $(-30, 0)$, and $(-32, 5)$. Note the close arrangement of two sources. Both algorithms process 20 frequency bands in the approximate frequency range 7.8 kHz – 8.6 kHz, so that the wavelengths are between 2 and 3 times the array spacing. The spectrum functions are evaluated on a $p \times p$ grid corresponding to azimuths and elevations between -45 and 45 degrees, where p is 128

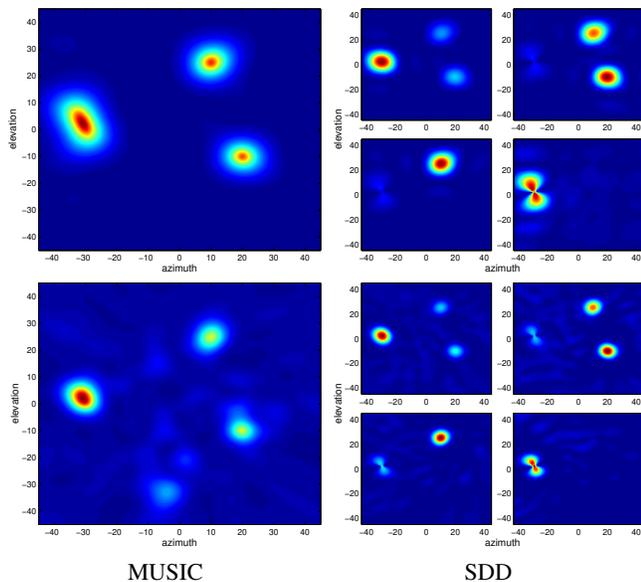


Fig. 2. The spectrum functions for experiments C and D displayed as contour plots on a dB scale. For SDD, the spectrum functions after 0, 1, 2, and 3 iterations are displayed left to right, top to bottom.

for MUSIC and 64 for SDD. For MUSIC, the sum of each individual band’s spectrum is used.

Experiment D: We use a more irregular array based on that of experiment E (Fig. 3(a)), with microphone spacing comparable to the grid array in experiment C. All other details of this experiment mirror that of experiment C.

Experiment E: As a test of SDD’s viability in practice, we use a 64-element array to record a moving source inside a room. 32 frames of these recordings were processed with SDD on 50 frequency bands in the 2.5 Hz–4 kHz frequency range. The SDD objective function was evaluated on a 32×32 grid over -45 to 45 degrees in azimuth and elevation. The array also recorded a video using a camera mounted in the array’s center; as ground truth, we compute incident angles using the video frames corresponding to the processed frames.

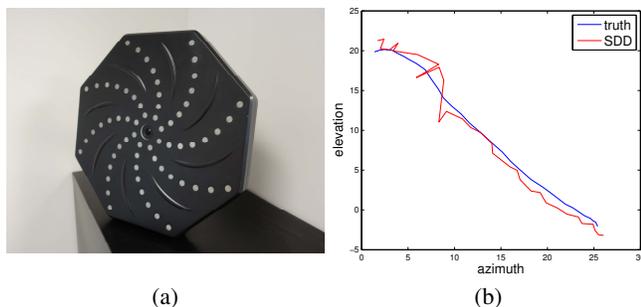


Fig. 3. The array used in experiment E (a), and parametric plots of the true and computed source directions in experiment E (b).

5. DISCUSSIONS AND CONCLUSIONS

Fig. 1 shows results of experiments A and B. From experiment A, we see that the peak of SDD is similar to that of MUSIC for a single-source recording. For multi-source recordings, the initial spectrum of SDD is much flatter in comparison to MUSIC, and has much wider peaks. However, note that these peaks are not the actual peaks from which the DOA are derived by SDD: only the strongest peak is used at the first step of the algorithm. Peaks which are flat or even

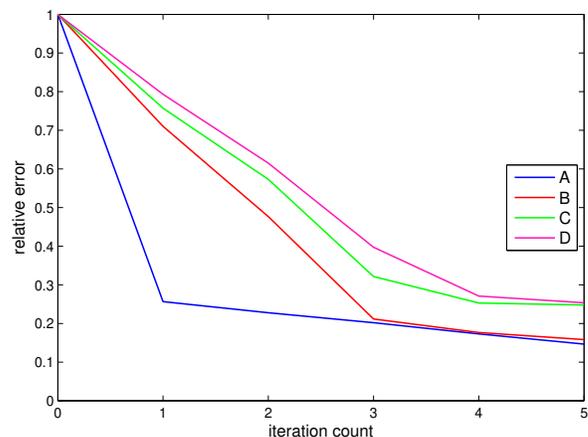


Fig. 4. The relative error after each iteration of SDD in all synthetic experiments.

invisible at the initial steps become stronger and sharper in consequent steps after stronger peaks are removed. Experiment B shows that the peaks of SDD at the first step of the algorithm are comparable in width to those of MUSIC when a single frame is used for covariance estimation.

Fig. 2 similarly shows that SDD was able to detect all four sources in experiments C and D. We also observe a noticeable difference in the MUSIC spectrums of these experiments: in experiment D, the MUSIC spectrum contains an extraneous peak near $(-5, -30)$. Since the MUSIC method in this case assumes a signal subspace of rank 4, we can infer that the method was unable to distinguish between the close sources and treated them as one source. In comparison, there are no such extraneous peaks in the MUSIC spectrum for experiment C, suggesting that the method correctly accounted for all four sources. As SDD produced similar results in both experiments, this result suggests that SDD is less dependent on array geometry than MUSIC.

Fig. 3(b) plots the principal source directions computed by SDD and those computed as ground truth as a pair of curves parameterized by frame number. The curve for SDD travels in roughly the same path as the ground truth, albeit with some fluctuations. Such a result suggests that SDD is also applicable in real environments.

We now analyze the relative error computed after each iteration of SDD in all synthetic experiments. As seen in Fig. 4, if we set the error tolerance as ≈ 0.3 , the method yields the correct number of sources in all experiments. Note that this tolerance is close to the relative amplitude of the -10 dB microphone self-noise. Thus for optimal performance we should set an error tolerance proportional to the magnitude of measurements not accounted by the SDD model.

This algorithm is similar to classical MUSIC [1] and its extensions [4, 6]. The latter paper considers extensions similar to that considered here: multi-frequency sources. However, no decomposition is performed there; instead, there are strong assumptions on the frequency content and numbers of sources. In contrast, our algorithm is general.

An interest of ours is to pursue a representation that combines the leading planewaves/sources and the ambient field. To achieve this, the current algorithm would have to be extended to include near sources, and finally the residual after SDD/source estimation would be represented via a low-order ambisonics representation. Other possible uses include source localization/separation. Future work will look at obtaining real time implementations, extend the algorithm to arrays on baffled objects, and further characterization of this algorithm.

6. REFERENCES

- [1] R. Schmidt, "Multiple emitter location and signal parameter estimation," *IEEE Transactions on Antennas and Propagation*, vol. 34, no. 3, pp. 276–280, Mar 1986.
- [2] A. O'Donovan, R. Duraiswami, J. Neumann. "Microphone arrays as generalized cameras for integrated audio visual processing," IEEE Conference on Computer Vision and Pattern Recognition, 2007. CVPR'07, 1-8
- [3] J. H. DiBiase, H. F. Silverman, and M. S. Brandstein, "Robust localization in reverberant rooms," in *Microphone Arrays: Signal Processing Techniques and Applications*, M. S. Brandstein and D. B. Ward, Eds., pp. 157–180. Springer Berlin Heidelberg, Berlin, Heidelberg, 2001.
- [4] D. Kundu, "Modified MUSIC algorithm for estimation DOA of signals," *Signal Process.*, vol. 48, no. 1, pp. 85–90, Jan. 1996.
- [5] B. Rafaely, 2004. Plane-Wave Decomposition of the Sound Field on a Sphere by Spherical Convolution, *J. Acoust. Soc. Am.*, vol. 116(4), pp. 2149-2157.
- [6] T. Terada, T. Nishimura, Y. Ogawa, T. Ohgane, and H. Yamada, "DOA estimation for multi-band signal sources using compressed sensing techniques with Khatri-Rao processing." *IEICE Transactions on Communications*, vol. E97.B, no. 10, pp. 2110–2117, 2014.
- [7] D. N. Zotkin, R. Duraiswami and N. A. Gumerov. "Plane-Wave Decomposition of Acoustical Scenes Via Spherical and Cylindrical Microphone Arrays," *IEEE transactions on audio, speech, and language processing*. 20(1):2-2, 2010.