ROBUST BLIND SOURCE SEPARATION IN A REVERBERANT ROOM BASED ON BEAMFORMING WITH A LARGE-APERTURE MICROPHONE ARRAY


Dept. of Electrical Engineering, Princeton University, Princeton, NJ 08544
{jsanz, liechaoh, tmyo, rieutort, yingzheh, wagner, sturm, nverma}@princeton.edu

ABSTRACT
Large-Area Electronics (LAE) technology has enabled the development of physically-expansive sensing systems with a flexible form-factor, including large-aperture microphone arrays. We propose an approach to blind source separation based on leveraging such an array. In our algorithm we carry out delay-sum beamforming, but use frequency-dependent time delays, making it well-suited for a practical reverberant room. This is followed by a binary mask stage for further interference cancellation. A key feature is that it is fully “blind”, since it requires no prior information about the location of the speakers or microphones. Instead, we carry out k-means cluster analysis, to estimate time delays in the background from acquired audio signals that represent the mixture of simultaneous sources. We have tested this algorithm in a conference room (which we refer to as the “acoustic path”), making such a frame well-suited for time delay extraction. We use k-means clustering, an unsupervised classification technique, to identify a short (64 ms) frame at the beginning of the sound mixture in which only a single source is prominent, making such a frame well-suited for time delay extraction.

When developing an algorithm for isolating different sources in a practical room, known as the blind source separation (BSS) problem, one of the principal challenges is the unpredictability of the acoustic path. Not only is the path affected by reverberations with surfaces and objects in the room, but human sources can move. To solve BSS one approach is beamforming, which leverages the spatial filtering capability of a microphone array to isolate sources. Unfortunately, classical delay-sum beamforming is not well-suited to a practical room. This is because it uses pre-defined time delays that are independent of frequency between the microphones, with the aim of constructively adding the signal from a target source and destructively adding the signals from all interfering sources [3]. An alternative approach to BSS is to use algorithms based on a frequency domain implementation of independent component analysis (ICA), which typically exploit statistical independencies of the signals [4]. However, there are concerns about the robustness of these algorithms, especially in a reverberant room. This is due to the inherent permutation ambiguity of this approach, where after separation independently at each frequency, the components must further be assigned to the correct source. This necessitates an additional decision step [5].

Weinstein et al. [6] were able to isolate speech signals using conventional delay-sum beamforming, but had to utilize an array with over 1000 microphones to obtain acceptable results. Levi et al. continued to use conventional delay-sum beamforming, but incorporated a spectral subtraction step based on SRP-PHAT after beamforming, enabling an array with just 16 microphones [7]. Unfortunately, this approach is not blind, since it requires the location of the sources and microphones.

In this work we propose and demonstrate a beamforming-based algorithm for BSS, with the following main contributions:

1. We also use delay-sum beamforming, but unlike prior work we do not use a single time delay across all frequencies for a given microphone-source pair. Rather we use frequency dependent time delays. This is needed since reverberations from the surfaces in a practical room lead to multipath propagation, for which a linear phase model is inadequate [8].

2. We crucially differ from other beamforming attempts by being blind, requiring no prior information about the location of the sources or microphones. The only information our algorithm needs about the environment is the number of sources. Thus, we avoid time consuming and technically challenging location measurements [9]. Furthermore, we make no assumptions about the propagation of sound in a room. Rather, we extract time delays for each microphone-source pair on the fly from the sound mixture of simultaneous sources. This enables our algorithm to adapt to the unique acoustic properties of each room (e.g., size, reverberation time, placement of objects) and a change in location of the sources.

3. We apply our algorithm to experimental data from two adjacent linear arrays, measured in a conference room: (1) an array of
2. ALGORITHM

2.1. Problem Setup

The array consists of $M$ microphones, which separate $S$ simultaneous sound sources, $x_s(t)$. The sound recorded by each microphone, $y_m(t)$, is determined by the room impulse, $h_{m,s}(t)$, between each source and microphone:

$$y_m(t) = \sum_{s=1}^{S} x_s(t) * h_{m,s}(t).$$

We designate one of the microphone channels as a reference, $ref$, and express the signal recorded in the time-frequency domain at this reference microphone, for frequency $\omega$ and frame $L$ as:

$$Y_{ref}(\omega, L) = \sum_{s=1}^{S} X_s(\omega, L)|H_{ref,s}(\omega)|e^{j\omega T_{ref,s}(\omega)}$$

(2)

where $H_{ref,s}(\omega) = |H_{ref,s}(\omega)|e^{j\omega T_{ref,s}(\omega)}$ is the room impulse response in the frequency domain, and $T_{ref,s}(\omega)$ is the time delay between the reference microphone and a source $s$. Our objective is to recover each source $s$ at the reference microphone, as if it were recorded with the other sources muted:

$$X^*_s(\omega, L) = X_s(\omega, L)|H_{ref,s}(\omega)|e^{j\omega T_{ref,s}(\omega)}.$$  

(4)

2.2. Beamforming with Frequency Dependent Time Delays

The first step of our algorithm is delay-sum beamforming. During this step, for a given source we time align all microphone signals with respect to the reference microphone and sum them:

$$\hat{X}^*_s(\omega, L) = \sum_{m=1}^{M} Y_m(\omega, L)e^{-j\omega D_{ms}(\omega)}$$

(5)

where $D_{ms}(\omega)$ is the time delay between the reference and each microphone. In this way we constructively sum the contributions from the source we want recover over all microphones, and attenuate the other sources though destructive interference.

In classical delay-sum beamforming, $D_{ms}(\omega)$ is treated as a constant, frequency-invariant value, such as found in anechoic conditions [3]. Instead, this implementation takes into account multipath propagation of sound in a reverberant room, which has the effect of randomizing the phase spectrum of the room impulse response [8].

2.3. Binary Mask

To further suppress interfering sound sources, a binary mask, $M_s(\omega, L)$, is applied to the output of the delay-sum beamformer:

$$\hat{X}^*_s(\omega, L) = \hat{X}^*_s(\omega, L)M_s(\omega, L).$$

(6)

When constructing a binary mask, frequency bins are assigned a value of 1 if they meet the following criterion, otherwise they are assigned a value of 0:

$$\frac{|\hat{X}^*_s(\omega, L)|}{\max(|\hat{X}^*_s(\omega, L)|, |\hat{X}^*_s^2(\omega, L)|, \cdots, |\hat{X}^*_s^N(\omega, L)|)} > \alpha$$

(7)

where $\alpha$ is a constant threshold value that is experimentally tuned. After applying the binary mask, the inverse FFT is taken of each frame to recover the time domain signal, and successive frames are concatenated using the standard Overlap-Add method.

2.4. Time Delay Estimates Based on k-Means Clustering

Time delays between the reference and other microphones, can be estimated by making each source play a test sound one-by-one in isolation. A frame from the test sound, such as speech or white noise with the desired spectral content, can be used to find the time delays:

$$D_{ms}(\omega) = T_{ms}(\omega) - T_{ref,s}(\omega, L) = \frac{1}{2\pi f} \angle X^*_m(\omega, L) - \angle X^*_s(\omega, L) = \frac{\phi_m(\omega, L)}{2\pi f}$$

(8)

where $f$ is the frequency and $\angle X^*_m(\omega, L)$ is the phase of a frame from the desired source recorded at microphone $m$. We replace this calibration procedure by estimating the time delays directly from the signal when all sources are playing simultaneously. We are able to achieve this by using a standard implementation of k-means clustering based on euclidean distance [11]. We set the number of clusters, $k$, to be equal to the number of sources, $S$. A feature vector is extracted for each frame, which consists of the phase difference, $\phi_m(\omega, L)$, between a given microphone, $m'$, and the reference at the $N$ frequencies of interest:

$$\phi_m(\omega, L) = \theta_{\omega_1}, \theta_{\omega_2}, \cdots, \theta_{\omega_N}$$

(9)

with $\theta$ taken to be in the range $[0, 2\pi)$.

Our intent is not just to classify each frame as belonging to a given source, since many frames have spectral content from multiple sources, which would lead to poor time delay estimates. Rather we
want to identify the best possible frame from which to derive the
time delays. To identify these frames we calculate the silhouette
[12], \( s(L) \), for every feature vector, and choose the frame with the
highest value:

\[
s(L) = \frac{b(L) - a(L)}{\max(b(L), a(L))}
\]

where \( a(L) \) is the mean distance between the feature vector from the
frame with index \( L \) and all other feature vectors assigned to the same
cluster. Then, the mean distances to the feature vectors correspond-
ing to all other clusters are also calculated, and the minimum among
these is designated as \( b(L) \). The value of \( s(L) \) is bounded between
\([-1, 1]\], and a larger value indicates it is more likely a feature vector
has been assigned to an appropriate cluster.

3. EXPERIMENTAL RESULTS

3.1. Setup Conditions

Experiments were carried out in conference room, as shown in
Figure 2, playing both two (B and C) and four (A, B, C and D) si-
nultaneous sound sources from a loudspeaker (Altec ACS90). Table
1 has a summary of experimental conditions. The two linear arrays
were mounted horizontally, with a PVDF microphone approximately
3 cm above a corresponding electret microphone; thus, allowing us
to directly compare the performance of the two arrays. Each array
used different elements: (1) Commercial omnidirectional elec-
tret capsules (Primo Microphone EM-172); (2) LAE microphones,
which are based on a flexible piezoelectric polymer, PVDF, and are
fabricated in-house. Figure 3 shows the frequency response of both
types of microphones, including the non-idealities of LAE micro-
phones arising due to the fabrication methods which lead to their
large-area, thin, and flexible form factor e.g. reduced sensitivity, a
non-flat response and large variations across elements.

To assess the performance of our algorithm we used two metrics:
(1) Signal-to-Interferer Ratio (SIR) calculated with the BSS Eval
Toolbox [14] [15]; (2) PESQ using the clean recording from the TSP

Number of Sources \( S = 2 \) (B.C) and \( S = 4 \) (A, B, C, D)
Number of Microphones \( M = 16 \)
Microphone Pitch 15 cm (total array width = 2.25 m).
Source Signals 12 Harvard sentences from the
TSP database [13] (Duration = 30 s).
Sampling Rate 16 kHz
Reverberation Times \( T_{60} = 350 \) ms
Window Type Hamming
STFT Length 1024 samples (64 ms)
STFT Frame Shift 256 samples (16 ms)
Reference Microphone Located at center of linear array.
Threshold for Binary Mask \( \alpha = 1.4 \) (see Equation 7).

3.2. Time Delay Estimator Performance

We compared the performance of our algorithm using time delays
extracted under two conditions: (1) White Gaussian noise, which
was played by each speaker one-at-a-time, before the simultaneous
recording, and (2) from a single frame of simultaneous speech that
was selected by our k-means-based silhouette criterion. It should be
noted that to improve the estimate when extracting the time delays
from white noise, the phase difference in Equation 8 consisted of the
circular mean [17] calculated from 50 successive frames.

To identify the best frames for time delay extraction, we imple-
mented k-means with 312 features vectors. Each feature vector was
extracted from a different frame (frame length = 64 ms, frame shift
=16 ms) taken from the first 5 s of the recording with the simultane-
ous sources. We used a total of 160 features, corresponding to the
phase difference between the closest adjacent microphone and the
reference microphone for each frequency bin between 500 Hz and
3000 Hz.

After k-means, the silhouette was calculated for all 312 feature
vectors in order to select a feature vector per source for extracting
time delays. Figure 4 validates the use of the silhouette as a metric
for selecting a frame to use for time-delay extraction (calculated after
the beamforming stage, using time delays extracted from the feature
vector, for two simultaneous sources). Figure 5 shows a compar-
ison, for two representative microphones in the array, of the phase
delays estimated using white noise played in isolation versus those
estimated from frames selected based on the silhouette. Good agree-
ment is observed. Below we also compare the performance of our
algorithm when using time delays from white noise and k-means. In
most experiments there is only a small performance degradation for
k-means, highlighting its effectiveness for enabling BSS.

3.3. Overall Algorithm Performance

A lower limit on performance is given by calculating the SIR and
PESQ at the reference microphone before any signal processing. An
the same experiment with the LAE microphone array and find that
the effectiveness of our proposed algorithm. A mean PESQ improve-
capsules, the PESQ is nearly the same as the upper limit (e.g. the
obtain sources.
but for four sources it sometimes fails to significantly enhance cer-
tain 16 microphone array) outperforms by a wide margin. On
for 2 sources, 4 microphones for 4 sources) our algorithm (using
the minimum number of microphones for IVA BSS (2 microphones
conventional BSS algorithm, we chose IVA BSS [10]. When using
binary mask, outperforms using only the beamforming stage.
shows how our algorithm, combining beamforming followed by a
anechoic recording that was inputted into the loudspeaker. In Figures
upper limit is given by the PESQ at the reference microphone when
only a single source is playing, using as a reference signal the clean
anechoic recording that was inputted into the loudspeaker. In Figures
6 to 9, we show that for all configurations our algorithm successfully
enhances speech, significantly increasing both SIR and PESQ. It also
shows how our algorithm, combining beamforming followed by a
binary mask, outperforms using only the beamforming stage.
To compare the performance of our algorithm with a modern,
conventional BSS algorithm, we chose IVA BSS [10]. When using
the minimum number of microphones for IVA BSS (2 microphones
for 2 sources, 4 microphones for 4 sources) our algorithm (using
the entire 16 microphone array) outperforms by a wide margin. On
the other hand when using IVA BSS with the entire array and
selecting the best channels from the 16 separated channels it out-
putted, IVA BSS and our algorithm perform at a similar level. For
two sources IVA BSS performs slightly better than our algorithm,
but for four sources it sometimes fails to significantly enhance cer-
tain sources.
In Figure 6, when using two sources and the array with electret
capsules, the PESQ is nearly the same as the upper limit (e.g. the
sound played in isolation at the reference microphone), highlighting
the effectiveness of our proposed algorithm. A mean PESQ improve-
ment of 0.7 is obtained when comparing the blind algorithm (with
k-means delays) to the unprocessed signal. In Figure 7, we repeat
the same experiment with the LAE microphone array and find that

4. CONCLUSION
We develop a beamforming algorithm for blind source separation us-
ing a large-aperture microphone array. The algorithm estimates time
delays between each source and microphone from the sound mixture
of simultaneous sources, by using k-means cluster analysis to iden-
tify suitable frames for the estimate. This enables our algorithm to be
“blind”, since we do not require the location of the microphones and
sources, and can adapt to the acoustic properties of each room and
a change in location of the sources. We tested the algorithm using
both commercial electret and LAE microphone arrays, with two and
four simultaneous sources, and in all cases we obtained significant
improvements in speech quality, as measured with PESQ and SIR.
These improvements, combined with the simplicity of our algorithm,
makes it a strong potential candidate for a real-time implementation
for an embedded system.

Fig. 5. Comparison of phase for two representative microphones
extracted from white noise and k-means (a) Microphone 4, Source
B; (b) Microphone 12, Source C.

Fig. 6. Separating two sources with an array of electret microphones.

Fig. 7. Separating two sources with an array of LAE microphones.

Fig. 8. Separating four sources with an array of electret microphones

Fig. 9. Separating four sources with an array of LAE microphones.
REFERENCES


