ANGLE–OF–ARRIVAL (AOA) FACTORIZATION IN MULTIPATH CHANNELS

Yu-Lin Wei and Romit Roy Choudhury
Department of Electrical and Computer Engineering,
University of Illinois at Urbana-Champaign

ABSTRACT
This paper considers the problem of estimating $K$ angle of arrivals (AoA) using an array of $M > K$ microphones. We assume the source signal is human voice, hence unknown to the receiver. Moreover, the signal components that arrive over $K$ spatial paths are strongly correlated since they are delayed copies of the same source signal. Past works have successfully extracted the AoA of the direct path, or have assumed specific types of signals/channels to derive the subsequent (multipath) AoAs. Our method builds on the core observation that signals from multiple AoAs embed predictable delay-structures in them, which can be factorized through iterative alignment and cancellation. Simulation results show median AoA errors of $< 4^\circ$ for the first 3 AoAs. Real-world measurements, from a circular microphone array similar to Amazon Echo, show modest degradations. We believe the ability to infer even $K = 3$ AoAs can be helpful to various sensing and localization applications.

Index Terms— Angle-of-arrival, blind channel inference, cancellation, multipath, acoustics, localization.

1. INTRODUCTION
Angle of arrival (AoA) refers to the angle $\theta$ over which a signal arrives at a receiver. In reality, a transmitted signal bounces off multiple surfaces and arrives at the receiver over multiple AoA angles $\{\theta_1, \theta_2, ...\}$. This paper aims to infer the first $K$ AoA angles, $\theta_{1:K}$. Such capabilities can be helpful to various applications. For instance, a smart speaker like Amazon Echo may be able to infer the location of a user by reverse triangulating the AoAs of the voice [1, 2]. Self-driving cars may be able to infer the acoustic sounds from other cars and infer their presence even when they are not visible around the corners [3]. Multiple AoAs may also serve as priors to source separation algorithms [4, 5, 6], aiding in better initial conditions for convergence.

While a rich body of past work have explored this problem space [7, 8, 9, 10, 11], most have either focussed on optimizing the direct path AoA $\theta_1$, or have assumed properties such as impulse-like signals [12, 13], multiple uncorrelated sources [8], co-prime channels [14, 9], etc. This paper is an attempt to estimate $\theta_{1:K}$ in uncontrolled conditions, such as real-world multipath channels and unknown acoustic source signals. Our proposed technique may even extend to RF signals, discussed at the end of Section 3.

The key challenge in estimating $\theta_{1:K}$ emerges from the strong correlated-ness of the arriving signal components. This correlated behavior is an outcome of both the auto-correlation of the source signal (especially human voices), compounded by the multipath channel which essentially delays and attenuates each signal component. The net received signal is thus a mixture of delays. Since relative delays (across microphones) are central to AoA estimation, the mixture poses a non-trivial challenge. Observe that if the source signal is known [15], or even if the source signal exhibits low auto-correlation (e.g., white noise) [16], the problem gets simplified. With human voice signals, these are clearly not the case.

Our proposed algorithm, called iterative AoA (IAoA), leverages the delay structure embedded in the signal, caused by the AoAs and microphone array geometry. As a starting point, IAoA estimates the relative delays of the direct path, a relatively simple problem, and infers the first AoA, $\theta_1$. Now, these relative delays allow us to align the signals and subtract them, such that only the subsequent multipath components remain as residues. By further aligning such residues from different pairs of microphones, we define an objective function, which gets minimized at the optimal AoA, $\theta_2$. Continuing this iteration, i.e., re-aligning and canceling subsequent signal paths, gives us $\theta_3$, and so on. In fact, when more microphones are available, the AoAs may also be jointly inferred (albeit at a larger computation cost). Section 3 will present the algorithm in greater detail.

We evaluate IAoA through simulation and real-world experiments. The real experiments are performed in a student apartment using a 6-microphone array from SEEED [17], mounted on top of a Raspberry Pi. We placed the device at different locations and gave voice commands, such as “Hey Siri”. The measurement data were used to parameterize the simulations, which in turn sheds light on a wide parameter space. Simulation results show that $75^{th}$ percentile AoA error is $< 4^\circ$ for $\theta_2$, and $< 3^\circ$ for $\theta_3$. The error degradation is small at lower SNRs. The performance in real environments degrades slightly, however, remains useful for various practical applications. We believe additional ideas can be developed atop IAoA to further reduce the multi-AoA errors.
2. PROBLEM FORMULATION

We consider an acoustic receiver with \( M \) microphones, denoted \( m_i, i \in [1, M] \). We denote the unknown source signal as \( x \), the signal recorded by each microphone as \( y_i \), and the corresponding acoustic channel as \( h_i \). \( IAoA \)'s goal is to accept all \( y_i \) as input, and output a list of angles \( \theta_j, j \in [1, K] \), where \( \theta_j \) is the AoA of the \( j^{th} \) multipath, and \( K \) is the maximum number of AoAs we intend to detect.

**Assumptions:** We allow the following assumptions:
- There is only one sound source \( x \) in the environment.
- The sound source is static for the duration of the signal.
- The receiver has at least \( M = 4 \) microphones.

**Drawing on Blind Channel Identification:**

As mentioned earlier, AoAs manifest as relative delays across microphones. If the acoustic channel at each microphone is known, then these relative delays are easy to calculate. This reminds us of blind channel identification (BCI), a rich body of work \([18, 14, 19, 20]\) aimed at estimating channels \( h_i \) when the source signal \( x \) is unknown. The core idea in BCI is to find the channels that align and cancel the received signals, i.e., find \( h_i, h_j \) s.t. \( y_i \ast h_j - y_j \ast h_i = 0 \). This requires the algorithm to estimate the amplitudes and channel taps \( \langle a_j, t_j \rangle \) at a reference microphone \( m_1 \), as well as the relative delays of the channel taps between \( m_1 \) and \( m_2 \). Note that these relative delays are a function of AoA, hence denoted as \( \Delta_{m_2}(\theta_j) \). Fig. 1 illustrates the case for 2 microphones.

![Fig. 1. Channel taps and their relation to AoA.](image)

Building on this, JADE \([21]\) jointly estimates the delay and AoA of each multipath with signal space analysis, and VoLoc \([2]\) iteratively solves the amplitude, delay, and AoA of each multipath. Authors in \([14]\) also show that sub-space methods converge when channels and source signals satisfy co-prime and linear complexity properties.

**Issues:** Existing solutions search over a large parameter space, \( \langle a_j, t_j \rangle \) and \( \Delta_{m_2}(\theta_j) \), leading to excessive computation and difficulties in convergence. Moreover, estimates of \( t_j \) are error prone because the microphones (in practical devices like Amazon Echo) are separated by less than 5cm. Thus, even small errors in these delay estimations contribute to large AoA error. This calls for sub-sample accuracy with BCI algorithms, which adds to the challenge in practical settings.

In contrast, AoA estimation does not need the \( t_j \) estimates, since we only rely on \( a_j \) and \( \Delta_{m_2}(\theta_j) \). This allows us to align and cancel each path successively, resulting in efficient AoA estimation. Finally, existing algorithms try to solve for all paths using the same microphone pair; additional microphone pairs only help increase the channel diversity and suppress noise. In contrast, we can combine the residue vector from different microphone pairs to cancel out the different paths. The details follow.

3. THE ITERATIVE AOA (IAoA) ALGORITHM

3.1. Intuition

The high level idea in \( IAoA \) is to align the channel taps associated to \( \theta_1 \) for any pair of microphones, say \( \{m_1, m_2\} \), and cancel them leaving only the subsequent taps \( \{\theta_2, ..., \theta_K\} \) as residue. The same can be obtained from another pair, say \( \{m_1, m_3\} \). Now, across these two residues from the two pairs, we intend to align and cancel the taps associated to \( \theta_2 \), requiring us to search on \( \Delta_{m_2}(\theta_2) \). We continue this iteratively till \( \theta_K \), requiring one additional microphone-pair for every new iteration. Observe that this relieves us from estimating \( t_j \) in each successive step – the key advantage with \( IAoA \). Figure 2 shows the algorithmic flow (and will become clear later).

3.2. Primitives

Observe that direct path AoA, \( \theta_1 \), is computed from a microphone pair \( \{m_1, m_2\} \). For \( \theta_2 \), we need two microphone pairs, say \( \{m_i, m_j\} \) and \( \{m_u, m_v\} \). In general, \( \theta_j \) needs to pick two groups of microphones from the \((j-1)th\) estimation residues. To this end, we define a hierarchy of microphone groups. For the \( \theta_j \) estimation, we need the \( j^{th} \) level microphone group, denoted as \( G_j \). We have:

\[
G_1 = \binom{M}{2}, \quad G_j = \left( \frac{G_{j-1}}{2} \right) \forall j > 1
\]

where \( M = \{m_i\} \forall i \in [1, M] \) is the set containing all the microphones. \( G_1 \) denotes all the microphone pairs, \( G_2 \) denotes pairs of microphone pairs, and \( G_j \) is composed of two lower level microphone groups. Of course, microphone groups allow the same microphone to be chosen into different pairs, e.g., \( \{m_1, m_2\}, \{m_1, m_3\} \in G_2 \).

3.3. The \( IAoA \) Algorithm

**Step 1: Find direct path AoA \( \theta_1 \), then align and cancel:** For every microphone pair \( g = \{m_i, m_j\} \) in \( G_1 \), \( IAoA \) estimates \( \theta_1 \) from cross correlation.

\[
\hat{\theta}_1 = \arg \max_{\theta} \sum_{i,j} corr(Y_i \times z^{-\Delta_{m_j}(\theta)}, Y_j \times z^{-\Delta_{m_i}(\theta)})
\]

where \( corr \) is the correlation coefficient function, \( Y_i = FFT(y_i) \), and \( \Delta_{m_j}(\theta) \) is the relative signal delay at \( m_j \) (w.r.t. reference \( m_1 \)) for AoA \( \theta \). \( z^{-d} \) denotes the \( z \)-transform operation of delaying signal by time \( d \). Now, aligning on \( \theta_1 \), we calculate the cancellation residue \( R(\{m_i, m_j\}) \) w.r.t. \( \theta_1 \).

\[
R(\{m_i, m_j\}) = Y_i \times z^{-\Delta_{m_j}(\theta_1)} - Y_j \times z^{-\Delta_{m_i}(\theta_1)}
\]
Step 1: Find $\theta_1$, align and cancel

Step 2: Calculate the residue channel

Step 3: Find $\theta_2$ using align and cancel

3.4. Some Comments on the Algorithm

- Observe that the align and cancel method is correct by design, i.e., it eliminates one channel tap at each iteration, leaving a signal residue that only contains subsequent channel taps. Failures occur when AoAs are angularly close, or when later-arriving signal paths are much stronger than earlier paths (causing $\theta_2$ to be estimated as $\theta_3$ in the second iteration, and $\theta_3$ estimated as $\theta_2$ in the third iteration). To alleviate such failure cases, we propagate the top $L$ $\theta$'s to the next iteration (instead of the best). This implies $\theta_2$ and $\theta_3$ are jointly estimated from the third iteration, where $\theta_3$ is searched over all angles while $\theta_2$ is searched only over $L$ values. Accuracy improves in many cases. The maximum number of AoAs we can estimate is $K = (M - 1)$. When excess microphones are available compared to a desired $K$, it is possible to either combine multiple microphone pairs to suppress noise, or use a smaller subset for speed-up. $IAoA$ applies to RF signals without modification. In fact, RF signals travel much faster than acoustics, so searching for taps $t_j$ must be performed at a much finer granularity. Since $IAoA$ bypasses this search, the benefits are greater. $IAoA$ can be executed on 3D AoAs as well by extending the correlation over 3D angles.

4. EVALUATION

4.1. Simulation Setup and Results

We first simulate $IAoA$ in MATLAB, with $M = 6$ microphones, and a room impulse response generated for a 5m by 5m room. The receiver location is varied with 1m granularity, while the transmitter is always 1m away but at varying angles (with 30° granularity). We aim to estimate $K = 3$ AoAs in this paper. For this, we run $r = 3$ iterations, factoring out $\theta_j$ at the end of the $j$th iteration. However, we also show the case where $\theta_3$ is estimated from $r = 2$ iterations, by picking the $\theta$ that gives the second-largest correlation peak.

Figure 3(a) shows the CDF of the error in estimating $\theta_2$, for different number of iterations $r$, and SNR. The median error for ($r = 2, SNR = 10dB$) is 1.68°. For ($r = 3, SNR = ...
the error reduces to 0.68°. The improvement occurs because the 3rd path contributes to θ2’s error in r = 2, but modeling both paths in r = 3 reduces error.

Fig. 3. Simulation CDF plot of (a) θ2 (b) θ3 estimation error with different number of iterations (denoted r) and SNRs.

Figure 3(b) shows the CDF of the error in estimating θ3. The median for (r = 2, SNR = 10dB) and (r = 3, SNR = 10dB) are 4.59° and 2.11°, respectively. Of course, r = 3 outperforms r = 2 because the second-best choice in r = 2 does not cancel the 2nd path’s impact on θ3 estimation.

Figure 4 shows AoA errors in a room (each point on the heatmap corresponds to the receiver location and the color at that location represents the median AoA error from different transmitter positions). Evidently, IAoA suffers when the receiver is near the center of the room where the wall-reflected paths are weak. The direct path is much stronger, hence small cancellation error from the direct path affects the estimation of θ2 and θ3. Also, θ1 and θ2 are often very similar since θ2 arrives after reflection from the wall behind the transmitter; θ3 does not suffer this problem, hence often performs better.

Fig. 4. Median error in the room: (a) θ2 (b) θ3 with r = 3.

4.2. Real-world Experiment

Figure 5 shows our receiver device and an example experiment setup. Our platform is a 6-microphone circular array from SEEED ReSpeaker [17], mounted on a Raspberry Pi4 (64 GB) [22] (we cannot use off-the-shelf Amazon or Google devices since they do not export the raw audio signals). The device is placed in a bedroom of dimensions 2.61m by 2.67m, around 0.5m away from two walls. A volunteer is asked to speak the wake word “Siri” from different room locations.

Figure 6(a) shows the error CDF when estimating θ2. The median error for r = 2 and r = 3 are 91.80° and 28.21° respectively. Observe that r = 3 considerably outperforms K = 2 because the receiver is placed in the corner, hence the reflections from two walls – the 2nd and 3rd multipath – have similar amplitudes. This shows the value of jointly modeling the 2nd and 3rd path to approximate the correct AoAs.

Fig. 6. Real-world CDF plot of (a) θ2 (b) θ3 estimation with different iterations.

Figure 6(b) shows the CDF of the third multipath’s AoA estimation error, θ3. The median error for r = 2 and r = 3 are 19.45° and 7°, respectively. Again r = 3 is better because it jointly models the 2nd and 3rd paths. Note that compared with θ2, θ3 estimation has an appreciably lower error. This is because there are some test cases where the user stands close to the wall, causing the 2nd path to be angularly close to the direct path (θ2 ≈ θ1). Hence, most of the 2nd path signal get canceled during the direct path cancellation. As a result, in the 2nd iteration, IAoA will align on θ3 instead of θ2.

Computation time: We evaluate IAoA on a desktop computer equipped with Intel i7-4780, 3.7GHz. The median completion times for r = 2 and r = 3 are 2.9 and 30 seconds.

5. CONCLUSION

We develop IAoA, an algorithm that factors out multiple AoA angles θ1:K from a microphone array. The technique holds under tenable assumptions and performs reasonably well even in real-world experiments. We believe IAoA can benefit several applications, including localization, source separation, and environment sensing in self-driving cars.
6. REFERENCES


