Localizing Multiple Acoustic Sources with a Single Microphone Array

Weiguo Wang, *Student Member, IEEE,* Jinming Li, *Student Member, IEEE,*
Yuan He, *Senior Member, IEEE,* Yunhao Liu, *Fellow, IEEE*

Abstract—The ability to localize acoustic sources can greatly improve the perception of smart devices (e.g., a smart speaker like Amazon Alexa). In this work, we study the problem of concurrently localizing multiple acoustic sources with a single smart device. Our proposal called *Symphony* is the first complete solution to tackle the above problem, including method, theory, and practice. The method stems from the insight that the geometric layout of microphones on the array determines the unique relationship among signals from the same source along the same arriving path. We also establish the theoretical model of *Symphony*, which reveals the relation between localization performance (resolution and coverage) and impacting factors (sampling rate, array aperture, and array-wall distance). Moreover, the ability to separate and localize multiple sources is also studied theoretically and numerically. We implement *Symphony* with different types of commercial off-the-shelf microphone arrays and evaluate its performance under different settings. The results show that *Symphony* has a median localization error of 0.662 m.

Index Terms—Localization, Acoustic, Microphone Array, Direction of Arrival

1 Introduction

Smart devices with sound recognition are now proliferating in our daily life. For example, smart speakers like Amazon ECHO, Google Home, Apple HomePod, and Alibaba Tmall Genie support various attractive applications, including voice control of home appliances, man-machine dialogue, and entertainment center.

With the fast development of smart home and office applications, there is an increasing need for acoustic source localization on smart devices. Whether the acoustic sources can be localized largely affects the capability and quality of the smart device's interactive functions, which include but are not limited to the following cases: (1) The ability of localization enables a smart speaker to process voice commands with user location awareness. When the user is lying in bed and says 'Turn on the light', the smart speaker can smartly switch off the ceiling lamp and turn on the bedside lamp, if the user's (namely the acoustic source) location is provided. (2) Localizing the acoustic source enables a smart device, e.g. the smart safeguard device, to better perceive the real situation. For example, the device may remind the parents of possible danger when it hears abnormal sounds of windows or doors from the baby's room. (3) Knowing the source location helps to authenticate voice commands. Recent studies have uncovered the vulnerabilities of smart speakers against inaudible and malicious voice commands [1], [2], [3], [4]. To defend against these threats, the smart speaker can leverage the knowledge of voice location to accept only the commands that originate from the real locations.

Conventional approaches of acoustic source localization require the deployment of multiple distributed microphone arrays. Based on the estimation of the source's time difference of arrival (TDOA) or direction of arrival (DoA) at the arrays, the source can be localized via triangulation [5], [6], [7], [8], [9], [10], [11]. However, those solutions cannot be applied to localization with a device like the smart speaker, which is usually equipped with only a single microphone array.

Acoustic source localization with a single array is a non-trivial problem. Note that the typical size of a microphone array is several centimeters at most, which is negligible with respect to the distance between the source and the array. As a result, the acoustic signal's propagation rays to the microphones are nearly parallel. Due to limited spatial resolution (array size or aperture) and temporal resolution (sampling rate of the microphone), a commercial array cannot distinguish DoAs of nearly parallel rays. This is the so-called far-field effect.

Exploiting the multi-path propagation paves a way to tackle the above problem, however, only in the scenario of localizing a single source. In addition to the line-of-sight path (denoted as LOS), VoLoc [12] leverages an additional arriving path by exploiting the nearby wall reflection (denoted as ECHO), and then localizes the far-field source after estimating DoAs of LOS and ECHO.

It is worth noticing that VoLoc assumes there is only one source in the sound field, which largely restricts its applicability in the real world. Usually, there are multiple acoustic sources in practice. For example, in the home environment, there may be family voices, television, washing machine, and microwave oven. These sources, including the voice commands, will interfere with each other, making it very difficult for VoLoc to localize them.

• *W. Wang, J. Li, Y. He, and Y. Liu are with the School of Software, Tsinghua University, Beijing, China. E-mail: {wwg18, li-jm19}@mails.tsinghua.edu.cn, heyuan@mail.tsinghua.edu.cn, yunhao@greenorbs.com. Yuan He is the corresponding author.*

This work is supported by National Natural Science Fund of China No. U21B2007.

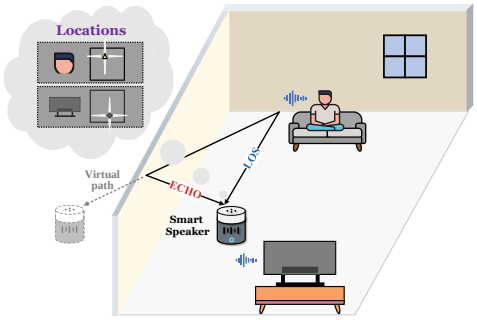


Fig. 1: An illustration of *Symphony*

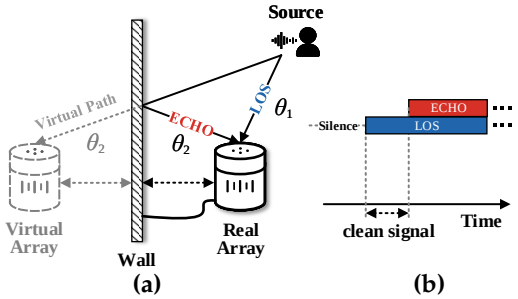


Fig. 2: (a) The virtual array is created by viewing the wall as a mirror. (b) Before the arrival of ECHO, there will be a short window of the clean signal.

Therefore, the ability of concurrently localizing multiple acoustic sources is quite appealing. Firstly, a smart speaker with such ability can naturally deal with inter-source interference: The smart speaker can intentionally view these interfering sounds as target sources and localize them as well. What's more, a smart speaker is able to interact with multiple users simultaneously, as long as they are in different positions. More generally, the perceptual ability of a smart speaker is further improved if the locations of multiple sources are known. Fig. 1 demonstrates an example: Given that the user and the television are localized, when the user says "Turn off the light", the smart speaker can infer that the user may want a better visual experience and then switch room lights to a Home-Theater mode [13], [14].

Localizing multiple acoustic sources with a single array is indeed a daunting task, with the following critical challenges: (1) The signals received at the array are a mixture of signals from multiple sources along different paths, making it extremely difficult to extract clean signals from any source. (2) The interference among the received signals blurs the relationship between signals and sources. It becomes very difficult to associate the DoA of a signal to its corresponding source, even if a DoA can be estimated. (3) The arriving paths of different sounds are diverse and unpredictable. The signals along those paths arrive after unpredictable delays. The exact arriving order of paths is unknown, thus hindering the discrimination of LOS and ECHO.

We propose *Symphony*, the first complete solution to localize multiple acoustic sources using a single microphone array. Our design stems from the following insights. (1) Although the propagation process of each source in the whole sound field is unpredictable, a microphone array will receive the signals of their arriving paths with predictable geometric patterns, determined by the pre-known layout of the array. (2) The coherence of sources exists not only among signals received by different microphones, but also among different arriving paths, LOS and ECHO.

Symphony exploits the redundancy in multiple microphone pairs with diverse spacing or (and) orientations to collaboratively estimate the DoAs of arriving paths. We design a novel algorithm that leverages intrinsic coherence among homologous paths to check whether two DoAs correspond to the same source. We find an interesting fact: the DoAs of LOS and ECHO should meet a certain criterion such that LOS and ECHO can intersect at a certain location. We formulate this criterion and apply it to discriminate LOS and ECHO. Our contributions are summarized as follows:

- To the best of our knowledge, *Symphony* is the first approach to localize multiple acoustic sources with a single array. The novel layout-aware design is inspired by the insight on the geometric redundancy residing in the microphone array, which is effective in resolving ambiguity induced by multiple sources.
- Symphony is a complete solution that addresses a series of technical challenges in multi-source localization. We estimate the DoAs of each path through a curve-fitting optimization process. To map DoAs to each source, we design a novel algorithm that exploits the coherence among homologous paths. We then formulate an intersection criterion to discriminate LOS and ECHO.
- We analyze how the localization resolution and coverage is affected by different factors, including sampling rate, array aperture, and array-wall distance. Moreover, the ability to localize and discriminate multiple sources is also studied theoretically and numerically.
- We implement *Symphony* with two commercial microphone arrays (4-mic linear array and 6-mic circular array), and evaluate its performance under different settings. The results show that *Symphony* has a median localization error of 0.662 m.

Roadmap. Section 2 introduces the background knowledge. Section 3 presents the overview. Section 4 introduces the propagation model of multiple sources. Section 5 to 8 elaborate on the design. Section 9 analyzes the localization performance. Section 10 presents the evaluation results. Section 11 discusses the related work. Section 12 and 13 discuss and conclude this work.

2 Primer

Far-Field Effect. The smart speaker typically holds a single microphone array. There is a critical barrier for a single microphone array to localize the source: the far-field effect. A source is considered to be in the far-field if [15]

$$L \geq \frac{2d^2}{\lambda}, \tag{1}$$

where L is the distance between the source and the array, λ is the wavelength of the arriving wave, and d is the inter-microphone distance.

Typically, a voice source is a far-field source to the microphone array: The fundamental frequency of the human speech (except singing) is less than 500 Hz, and the corresponding $\lambda > 0.66$ m. For an array with a size of 15

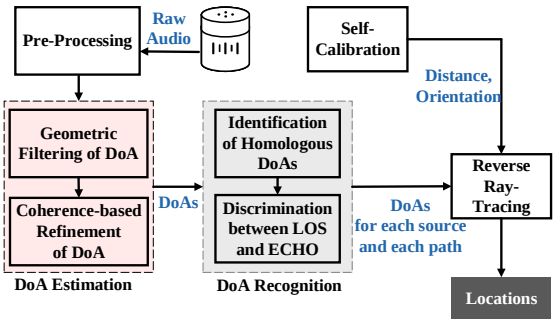


Fig. 3: *Symphony* System Overview.

cm, the source can be viewed as in the far field as long as it is 6.8 cm away from the source.

However, the far-field effect is not friendly to the localization task with a single array. Intuitively, if we can exactly obtain the DoAs of each propagation ray, the source can be easily localized as the intersection of these rays. Unfortunately, for the far-field sources, the propagation rays are nearly parallel, the DoAs of these rays are too close to be separated due to the limited spatial resolution.

Localization with a Single Array. VoLoc [12] leverages the nearby wall reflection to tackle the far-field effect, enabling a single array to localize the far-field source. The key idea is to create a virtual array by viewing the nearby wall as a mirror. Fig. 2(a) demonstrates this idea. The smart speaker is typically placed near a wall for power supply. If we view the wall as a mirror, a virtual array will appear behind this 'mirror'. In other words, besides the real array, we create an additional virtual array. The far-field source arrives at the real and the virtual array by the LOS path and the virtual path. Due to the relatively large distance between these two arrays, the LOS path and the virtual path will no longer be parallel, but have two distinguishable DoAs θ_1 and θ_2.

To estimate DoAs of LOS and ECHO, VoLoc takes advantage of pauses before a voice command. A voice command is usually preceded by silence. This means there will be a short time window during which the LOS signal is clean (see Fig. 2(b)). It is easy to derive the DoAs of LOS from the clean signal. Further, VoLoc views the clean signal as a template to model and cancel the ECHO path with appropriate alignment, and thus obtain the DoA of ECHO.

In practice, there are many other sources in a home, such as other people's voices, television, washing machine, microwave oven. These sources will interfere with voice commands in many scenarios. It is unlikely to obtain the clean signal of the targeted source. What's more, the microphones receive a mixture of signals from multiple sources along the LOS and ECHO paths, making it difficult to model and cancel other arriving paths. The application of VoLoc is largely restricted in the real world.

3 *Symphony* Overview

We propose a novel localization approach, *Symphony*, which enables a single smart speaker to localize multi-source concurrently. With such ability, the smart speaker can tolerate inter-source interference and, more importantly, improve its perceptual ability.

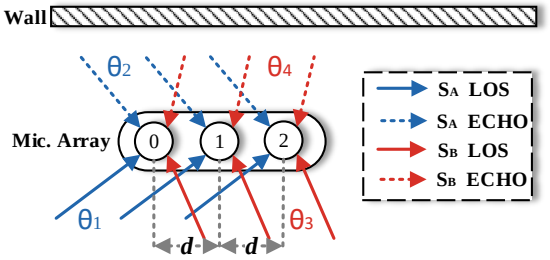


Fig. 4: The propagation model of two sources, S_A and S_B.

Fig. 3 shows the architecture of our system. To reverse ray-trace and localize multiple sources (Section 7), *Symphony* needs to obtain the DoA of LOS and ECHO for each source. *Symphony* adopts a two-stage scheme to obtain DoAs. (1) DoA Estimation (Section 5): *Symphony* uses geometric redundancy to produce high-resolution DoAs and leverages signal coherence to refine DoA results. (2) DoA Recognition (Section 6): *Symphony* then figures out which DoAs belong to the same source, and discriminates their types. To self-calibrate its own location, the smart speaker transmits probing pulses to measure the distance and orientation to the wall (Section 8).

4 Propagation Model

We begin with building the propagation model of multiple sources with multiple paths. The model mainly focuses on two paths: LOS and ECHO. This model will guide us in the estimation and the recognition of DoAs later.

Suppose two sources, S_A and S_B, are active simultaneously, and a linear microphone array with inter-distance d is recording the signals. In Fig. 4, two main paths of each source are illustrated: the direct path (LOS) and the wall reflection path (ECHO). For example, source S_A arrives at the array with DoA θ_1 directly, and its ECHO path arrives with DoA θ_2. We choose microphone M_0 as the reference. The signal received by microphone M_n at time t is

$$\begin{aligned} y_n(t) = {} & \alpha_A^{\text{los}} S_A\left(t - \tau_A^{\text{los}} - \mathcal{F}_n(\theta_1)\right) \\ & + \alpha_A^{\text{echo}} S_A\left(t - \tau_A^{\text{echo}} - \mathcal{F}_n(\theta_2)\right) \\ & + \alpha_B^{\text{los}} S_B\left(t - \tau_B^{\text{los}} - \mathcal{F}_n(\theta_3)\right) \\ & + \alpha_B^{\text{echo}} S_B\left(t - \tau_B^{\text{echo}} - \mathcal{F}_n(\theta_4)\right) \\ & + \text{other arriving paths of } S_A \text{ and } S_B, \end{aligned} \tag{2}$$

where α is the attenuation factor, $\tau_\bullet^{\text{los}}$ and $\tau_\bullet^{\text{echo}}$ are the times that the source $S_\bullet$ takes to arrive at the reference microphone M_0 along LOS and ECHO, respectively, and $\mathcal{F}_n(\theta)$ denotes the relative arriving delay between microphone M_0 and M_n for the signal coming from DoA θ. For a linear array with inter-distance d,

$$\mathcal{F}_n(\theta) = n\frac{d\cos\theta}{v}, \tag{3}$$

where v is the speed of sound. For simplicity, only LOS path and ECHO are considered in Eq. (2), and other paths are not specified. We discuss the reason at the end of this section.

GCC-PHAT (generalized cross-correlation with phase transform) is one of the most popular methods to estimate

DoA of wideband signals, and is able to suppress side-lobe effectively by discarding the amplitude of the cross-spectrum and keeping the phase [16], [17]. Consider two signals received by microphone M_n and M_m; the cross-correlation function (CCF)[1] between y_n and y_m is defined as

$$\mathrm{Cor}_{n,m}(\tau) = \mathrm{GCC}\left[y_n(t-\tau)\, y_m(t)\right], \tag{4}$$

where GCC$[\cdot]$ denotes the generalized correlation [16]. For the single-source case in free space, there is only one main correlation peak, and the time shift of the peak $\tau^* = \arg\max_\tau \mathrm{Cor}_{n,m}(\tau)$ captures the relative delay of the source to the microphone pair $\langle n, m\rangle$, which can then be used to calculate DoA based on Eq. (3).

However, the above method is infeasible in the case of multiple sources with multiple paths. Instead, to localize these multiple sources, *Symphony* needs to estimate DoAs of multiple paths of multiple sources, especially LOS and ECHO. Intuitively, instead of determining only one maximum correlation peak, we should determine multiple peaks to estimate multiple DoAs. After substituting Eq. (2) into Eq. (4) and assuming sources are mutually uncorrelated, we can find that there will be a series of correlation peaks. Table 1 lists these peaks' time shifts.

TABLE 1: Time shifts of correlation peaks between M_n and M_m.

Peak Type	Source	
	S_A	S_B
LOS-LOS	$\frac{d}{v}(m-n)\cos\theta_1$	$\frac{d}{v}(m-n)\cos\theta_3$
ECHO-ECHO	$\frac{d}{v}(m-n)\cos\theta_2$	$\frac{d}{v}(m-n)\cos\theta_4$
LOS-ECHO	$\frac{d}{v}(m\cos\theta_1 - n\cos\theta_2)$ $+\tau_A^{\mathrm{los}} - \tau_A^{\mathrm{echo}}$	$\frac{d}{v}(m\cos\theta_3 - n\cos\theta_4)$ $+\tau_B^{\mathrm{los}} - \tau_B^{\mathrm{echo}}$
ECHO-LOS	$\frac{d}{v}(m\cos\theta_2 - n\cos\theta_1)$ $+\tau_A^{\mathrm{echo}} - \tau_A^{\mathrm{los}}$	$\frac{d}{v}(m\cos\theta_4 - n\cos\theta_3)$ $+\tau_B^{\mathrm{echo}} - \tau_B^{\mathrm{los}}$

The peaks in Table 1 can be divided into two categories: **pure peaks** (LOS-LOS or ECHO-ECHO) and **hybrid peaks** (LOS-ECHO or ECHO-LOS). Such division is actually based on two basic facts:

- Firstly, let's zoom in on the array to observe the signal propagation on *a certain path*. We find that this path arrives at each microphone after different but short delays. These short delays are captured by pure peaks. As shown in Table 1, the time shifts of pure peaks depend only on the DoAs. This dependency implies that as long as we identify all pure peaks, we can estimate the DoA of each path. We elaborate on this in Section 5.
- Secondly, let's zoom out to the whole sound field to observe the signal propagation of *a certain source*. We can notice that this source has two main arriving paths, LOS and ECHO. Shortly after LOS arrives at the array, ECHO also reaches the array. The delay between LOS and ECHO is actually captured by hybrid peaks. Here, we point out that hybrid peaks will act as a bridge between the paths coming from the same source. We elaborate on this in Section 6.

1. Unless otherwise specified, CCF is a function of time shift, and the corresponding value denotes the correlation between the shifted versions of the inputs

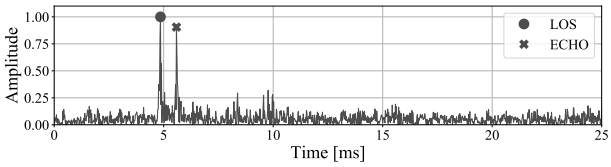


Fig. 5: Channel impulse response.

In summary, *Symphony* estimates DoAs based on pure peaks, and identifies which two DoAs belong to the same source based on hybrid peaks. We discuss the next question:

■ **The model considers only two paths, LOS and ECHO. What about the later arriving paths?** LOS is the first-arrival and strongest path because it traverses the shortest distance. Meanwhile, ECHO, which comes from the nearby wall reflection, traverses only a slightly longer distance than LOS and thus experiences nearly the same propagation attenuation as LOS[2]. On the other hand, the latter paths arrive after traversing much longer distances. This means the latter paths are considerably weaker than LOS and ECHO. We validate this by measuring a channel impulse response (CIR) in a living room. We place the array 30 cm away from the wall. As shown in Fig. 5, there are two distinct spikes with comparable amplitudes. These two spikes correspond exactly to LOS and ECHO. Their energy is dominant over all other paths. This explains why it is reasonable to consider only LOS and ECHO. In addition, Fig. 5 also shows that there is a distinguishable time interval (more than 1 ms) between LOS and ECHO, which means LOS and ECHO are not likely to be overlapped with each other in the time domain. This is because the sound speed is low, a certain distance difference (i.e., double array-wall distance) can result in a distinguishable time difference of arrival.

5 DoA Estimation

As mentioned in Section 4, DoAs can be calculated by the time shifts of pure peaks. In this section, we introduce how to identify all pure peaks among multiple correlation peaks. In Section 5.1, we exploit the geometric redundancy to filter out undesired peaks. However, this approach is not perfect due to limited spatial resolution. Therefore, in Section 5.2, we further leverage the intrinsic coherence to identify all pure peaks.

5.1 Geometry-Based Filtering of DoA

We now introduce our method to identify pure peaks (LOS-LOS and ECHO-ECHO) of multiple sources. Our insight is that instead of using a single microphone pair, we can take advantage of redundancy among multiple pairs with different layouts. Without loss of generality, we next use a uniform linear array (ULA) to introduce our idea. Note that

2. The energy loss resulting from wall reflection is negligible. Typically, more than 95% signal energy remains after reflection due to the high impedance mismatch between the air and the wall.

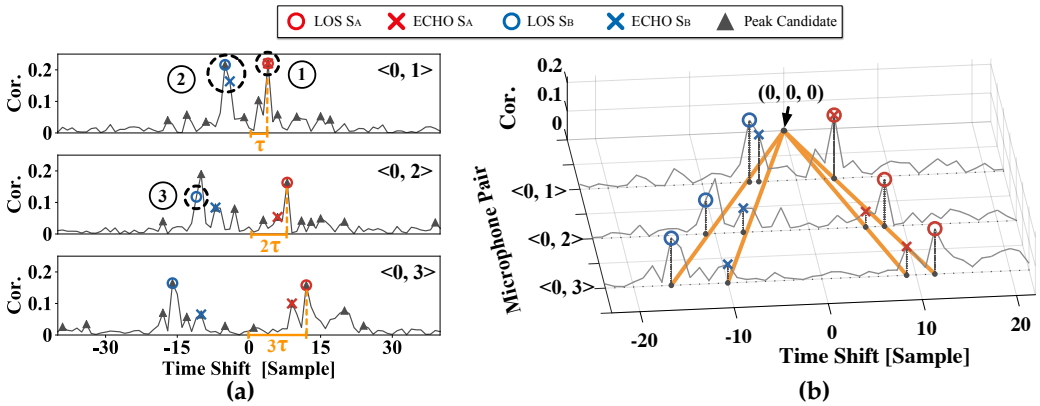


Fig. 6: **(a)** CCF between pairs $\langle 0,1\rangle$, $\langle 0,2\rangle$ and $\langle 0,3\rangle$. **(b)** Pure peaks across three pairs can fit a straight line passing $(0,0,0)$.

our idea can be applied to arrays with different layouts, including the circular array [18].

We observe that for microphone pair $\langle n,m\rangle$, the time shifts of pure peaks are directly proportional to the subtraction of two microphone serial numbers, $m-n$. If we revisit the cell $\langle$LOS-LOS, $S_A\rangle$ in Table 1, we will find that the relation between the time shift τ and the variable $m-n$ is a linear function: $\tau = k(m-n)$, where the slope $k = \frac{d}{c}\cos\theta_1$. This linear relation only holds for pure peaks, rather than hybrid peaks. We exploit this relation to find pure peaks.

We conduct the following proof-of-concept experiment to validate our idea. In a living room, we let two speakers simultaneously play two recorded voice commands (denoted as S_A and S_B) at different places. A uniform linear array with 4 microphones is placed 30 cm away from the wall to record signals. Fig. 6(a) shows the cross-correlation functions between three microphone pairs $\langle 0,1\rangle$, $\langle 0,2\rangle$ and $\langle 0,3\rangle$. In these plots, the markers with a triangular shape denote the local maximums of correlation, which are the pure peak candidates. The markers with the shape of circle and cross are the ground truths of pure peaks. The ground truths are obtained by inserting pseudo-random white noise (i.e., pre-known signals) immediately preceding voice commands. Fig. 6 (a) highlights the time shifts of pure peaks for LOS S_A (red circle), which change linearly with the subtraction of the microphone serial numbers, $m-n$.

To be clearer, we incorporate these three cross-correlation functions into the same coordinate system, as shown in Fig. 6 (b). It is very interesting to see that, if we sequentially connect the markers of the ground truths that correspond to the same path, the pure peaks can nearly form a straight line passing through origin $(0,0,0)$. This phenomenon motivates us to exploit this linear relation to find pure peaks. However, before introducing our method, it is worthwhile to analyze the following problems.

- **Problem 1: Peak Overlap.** Some markers of the ground truths are too close to be separated (e.g., ① in Fig. 6 (a)), or the marker is no longer the local maximum because it was suppressed by the adjacent peak (e.g., ②).
- **Problem 2: Peak deviation.** Some markers (e.g., ③) of the ground truths are not at the local maximum, and instead are one-sampling-point away from the nearest peak.

In fact, the above problems are caused by limited spatial resolution. Recall that the value we can measure is the time shift of peak, and the value we intend to obtain is the DoA θ. However, the sampling rate F_s limits the resolution of time shifts measured. The continuous DoA $\theta \in [0,\pi]$ is mapped into $2\lfloor \frac{(m-n)d}{v} F_s \rfloor + 1$ discrete bins. Such mapping introduces an additional conversion error, thus introducing the deviation between the ground truth and the peaks (e.g., ③). If two DoAs are so close that they fall into the same discrete bin, it is impossible for the array to separate them (e.g., ①). On the other hand, when $m-n$ increases, the number of discrete bins mapped also increases, thus providing higher spatial resolution. This explains why the markers of S_A no longer stay in the same bin in pair $\langle 0,2\rangle$, and also explains why Problems 1 and 2 are unlikely to occur for pair $\langle 0,3\rangle$.

Term	Brief description
$P_{n,m}$	The set of time shifts of peak candidates in pair $\langle n,m\rangle$.
$\tau_{n,m}$	The time shift of peak candidates $\tau_{n,m} \in P_{n,m}$.
$\boldsymbol{c_i}$	Possible combination of peak candidates across multiple pairs. For 4-mic ULA, $\boldsymbol{c_i} \in P_{0,1} \times P_{0,2} \times P_{0,3}$.
$w_{n,m}$	The penalty factor of pair $\langle n,m\rangle$, equal to $\lvert m-n\rvert$.

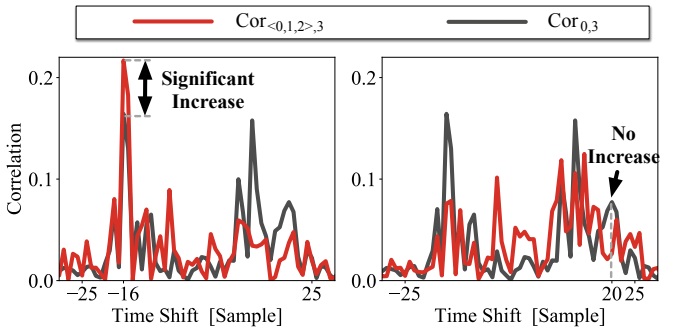
TABLE 2: Definition of Terminology.

i	$\boldsymbol{c_i}$ [sample]	$\mathcal{J}(\boldsymbol{c_i})$	k^*	pure peak	source
1	(4, 8, 12)	0.000	4.000	Yes	S_A
2	(6, 13, 20)	0.191	6.633		
3	(-1, -3, -5)	0.191	-1.633		
4	(-4, -7, -10)	0.191	-3.367	Yes	S_B
5	(-5, -10, -16)	0.258	-5.276	Yes	S_B
6	(2, 6, 9)	0.328	2.990	Yes	S_A
7	(2, 6, 7)	1.004	2.439		
8	(-4, -10, -13)	1.004	-4.439		

TABLE 3: The ranking list of $\boldsymbol{c_i}$ using metric $\mathcal{J}$ (Top 8).

The analysis above suggests that it is unrealistic to apply a strict criterion to directly identify pure peaks according to the linear relation. To tolerate such imperfections, we formulate DoAs estimation into a curve-fitting problem using a fitting metric. This metric evaluates how well peak candidates across each pair fit a curve. Formally, the metric is defined as follows (Table 2 defines the terminology):

$$\mathcal{J}(\boldsymbol{c_i}) = \min_k \frac{1}{|\boldsymbol{c_i}|} \sum_{\tau_{n,m}\in \boldsymbol{c_i}} w_{n,m}\left[k(m-n) - \tau_{n,m}\right]^2. \quad (5)$$



(a) Classify $\boldsymbol{c_5}$ = (-5, -10, -16) **(b)** Classify $\boldsymbol{c_2}$ = (6, 13, 20)

Fig. 7: Coherence-based refinement.

Intuitively, we select one candidate peak from each peak set for each microphone pair ($P_{n,m}$), and each possible selection constitutes a combination $\boldsymbol{c_i}$. The whole selection space is the Cartesian product of the peak sets of pairs ($P_{0,1} \times P_{0,2} \times P_{0,3}$). The metric $\mathcal{J}(\boldsymbol{c_i})$ evaluates the shortest distance between candidate peaks of $\boldsymbol{c_i}$ and the regression line $y = k(m-n)$. The $\boldsymbol{c_i}$ with the smaller $\mathcal{J}(\boldsymbol{c_i})$ is more likely to be the combination of pure peaks. As we discussed before, as $m-n$ increases, the resolution of pair $\langle n, m\rangle$ also increases, which means the pair with larger $m-n$ tends to have smaller error variance. This is a classic heteroscedasticity problem [19]. Following the idea of weighted least squares, we assign different pairs with different penalty factors $w_{n,m}$ inversely proportional to their error variances. In this way, we leverage results from pairs with different error variances, and can estimate DoA more accurately.

We apply the metric (Eq. (5)) to the previous proof-of-concept experiment. Specifically, we calculate $\mathcal{J}$ of each $\boldsymbol{c_i}$ in Fig. 6. Table 3 ranks each $\boldsymbol{c_i}$ in the ascending order of $\mathcal{J}(\boldsymbol{c_i})$. The slopes k^* of the lines fitted by $\boldsymbol{c_i}$ are also included. Based on the ground truths in Fig. 6, we mark the entries that belong to pure peaks. We can see that pure-peak entries get high ranks: first, fourth, fifth, and sixth, respectively. However, some entries not corresponding to pure peaks also get relatively high ranks: second and third, which may mislead the identification of pure peaks. The results show that this method can eliminate many ambiguities by ruling out low-ranking entries, but is incapable of identifying pure peaks confidently due to some outliers.

5.2 Coherence-Based Refinement of DoA

Next, we will refine the ranking results to finally determine pure-peak combinations.

Besides the geometric redundancy, the intrinsic coherence among signals received by microphones can also be exploited to identify pure peaks. Recall that pure peaks capture a certain path's relative delays among microphones. If we can compensate for these arriving delays based on pure peaks, signals received by microphones can be coherent with respect to a certain path. Based on this fact, we refine each entry $\boldsymbol{c_i}$ by checking whether the time shifts of $\boldsymbol{c_i}$ can make signals coherent with respect to a certain path. Next, we take a 4-mic linear array as an example to introduce our method. Algorithm 1 describes the refinement procedure.

In Algorithm 1, we expect that pure peaks in CCF increase. Specifically, if $\boldsymbol{c_i}$ is a pure-peak combination of a certain path, after aligning and averaging the first three signals y_0, y_1 and y_2 (Line 4 and 5), we can accurately compensate for the arriving delays of this path and constructively enhance the path. The enhanced version of the signal is denoted by $y_{<0,1,2>}$. Theoretically, when we correlate the enhanced signal $y_{<0,1,2>}$ with another signal y_3, the pure peak corresponding to this path (located at $c_i^{(3)}$) in CCF $\mathrm{Cor}_{<0,1,2>,3}$ will rise significantly, compared with the original CCF $\mathrm{Cor}_{0,3}$ between y_0 and y_3. If so (Line 7), $\boldsymbol{c_i}$ is identified as a pure-peak combination.

Algorithm 1: Refinement of Pure-Peak Entries

1 **Input:** signals y_0,y_1,y_2, and y_3 from a 4-mic array.
2 Compute CCF $\mathrm{Cor}_{0,3}$ between y_0 and y_3.
3 **for** *each* $\boldsymbol{c_i} = (c_i^{(1)}, c_i^{(2)}, c_i^{(3)})$ *in Table 3* **do**
4 Shift the signal y_1 and y_2 using the relative delays $c_i^{(1)}$ and $c_i^{(2)}$ to be in line with y_0.
5 Average the two shifted signals and the signal y_0, and denote the result as $y_{<0,1,2>}$.
6 Compute CCF $\mathrm{Cor}_{<0,1,2>,3}$ between $y_{<0,1,2>}$ and y_3
7 **if** $\mathrm{Cor}_{<0,1,2>,3}(c_i^{(3)}) - \mathrm{Cor}_{0,3}(c_i^{(3)}) >$ *Threshold* **then**
8 **Identify** $\boldsymbol{c_i}$ as a pure-peak combination.

We validate this idea in the previous proof-of-concept experiment. We compute $\mathrm{Cor}_{<0,1,2>,3}$ using the pure-peak combination $\boldsymbol{c_5}$ in Table 3, and the non-pure-peak combination $\boldsymbol{c_2}$. As expected in Fig. 7(a), the value of $\mathrm{Cor}_{<0,1,2>,3}$ at $c_5^{(3)}$ (sample -16) has a significant increase, while in Fig. 7 (b), no increase is observed at $c_2^{(3)}$ (sample -20). In summary, if there is an increase in $\mathrm{Cor}_{<0,1,2>,3}$ at $c_i^{(3)}$, $\boldsymbol{c_i}$ can be classified as a pure-peak combination.

6 DoA Recognition

After identifying all pure-peak combinations, we identify which two pure-peak combinations belong to the same source (Section 6.1). In Section 6.2, we determine their types.

6.1 Homologous Identification of DoA

A basic fact is that for a certain source, the ECHO path is a delayed version of the LOS path. These two paths are coherent because they come from the same source. *Symphony* exploits hybrid peaks to capture such inter-path coherence, thus identifying DoAs coming from the same source.

Without loss of generality, let's assume $\boldsymbol{c_i}$ and $\boldsymbol{c_j}$ belong to the same source, and correspond to LOS and ECHO respectively. Fig. 8(a) illustrates the definition of $\boldsymbol{c_i}$ and $\boldsymbol{c_j}$. Similar to Algorithm 1, after aligning and averaging the first three signals based on $\boldsymbol{c_i}$, we can obtain $y_{<0,1,2>}$ where the LOS signal is constructively enhanced, as shown in Fig. 8(b).

Note that the other received signal y_3 also receives the signals of LOS and ECHO of this source, which are both coherent with the enhanced LOS path of $y_{<0,1,2>}$. When we correlate $y_{<0,1,2>}$ with y_3, two peaks of CCF $\mathrm{Cor}_{<0,1,2>,3}$ will increase: (1) LOS-LOS, which is the correlation between the enhanced LOS in $y_{<0,1,2>}$ and the LOS in y_3; (2) LOS-ECHO, which is the correlation between the enhanced LOS

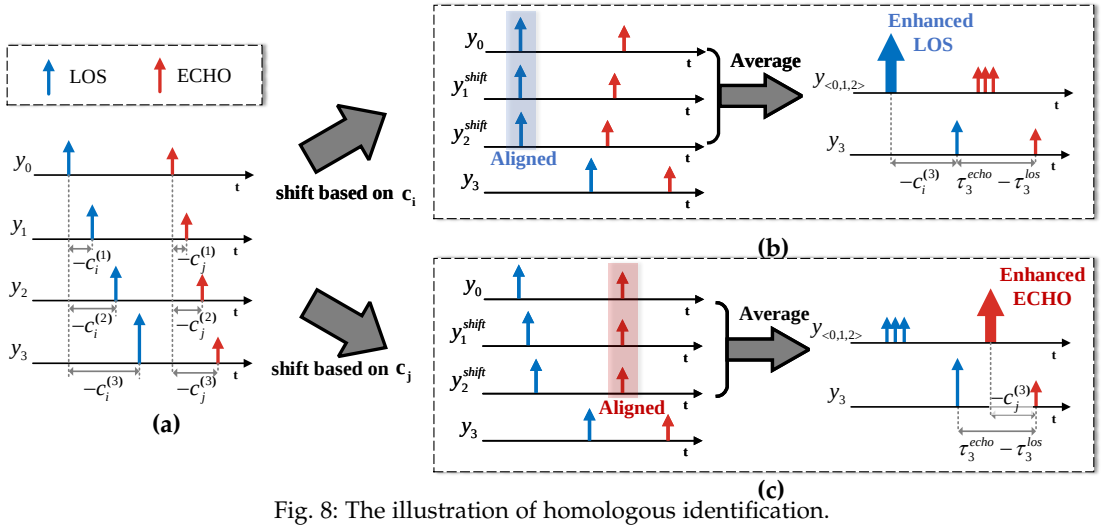


Fig. 8: The illustration of homologous identification.

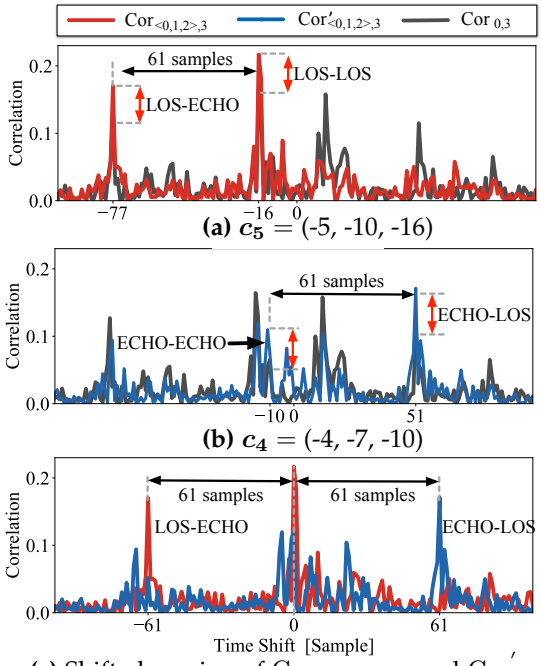


Fig. 9: Identify whether $\boldsymbol{c_5}$, $\boldsymbol{c_4}$ belong to the same source.

in $y_{<0,1,2>}$ and the ECHO in y_3. These two peaks are located at

$$\begin{cases} c_i^{(3)}, & \text{(LOS-LOS)} \\ c_i^{(3)} + \tau_3^{\text{los}} - \tau_3^{\text{echo}}, & \text{(LOS-ECHO)}. \end{cases} \tag{6}$$

where $c_i^{(3)}$ is the third element of $\boldsymbol{c_i}$, and τ_3^{los} and τ_3^{echo} denote the times when LOS and ECHO of the source arrive at microphone M_3, respectively.

Similarly in Fig. 8(c), if we use $\boldsymbol{c_j}$, which corresponds to ECHO of the source, to calculate the enhanced CCF (denote this CCF as $\mathrm{Cor}'_{<0,1,2>,3}$), we can also observe two peaks in CCF $\mathrm{Cor}'_{<0,1,2>,3}$ will increase: ECHO-ECHO and ECHO-LOS, which are located at

$$\begin{cases} c_j^{(3)}, & \text{(ECHO-ECHO)} \\ c_j^{(3)} + \tau_3^{\text{echo}} - \tau_3^{\text{los}}, & \text{(ECHO-LOS)}. \end{cases} \tag{7}$$

By comparing Eq. (6) and (7). we can find an interesting observation: the locations of hybrid peaks that get enhanced (i.e., LOS-ECHO in Eq. (6) and ECHO-LOS in Eq. (7)) are associated by a term $\tau_3^{\text{los}} - \tau_3^{\text{echo}}$. This is because the combinations $\boldsymbol{c_i}$ and $\boldsymbol{c_j}$ correspond to the same source, and thus both enhanced hybrid peaks capture the same arriving delay of the source between LOS and ECHO.

This association actually provides us an additional constraint to determine whether two pure-peak combinations $\boldsymbol{c_i}$ and $\boldsymbol{c_j}$ ($i \neq j$) belong to the same source. We take the following steps to apply this constraint:

1) Fetch $\mathrm{Cor}_{<0,1,2>,3}$ and $\mathrm{Cor}'_{<0,1,2>,3}$ which have already been computed in Algorithm 1 by using $\boldsymbol{c_i}$ and $\boldsymbol{c_j}$.
2) Shift $\mathrm{Cor}_{<0,1,2>,3}$ and $\mathrm{Cor}'_{<0,1,2>,3}$ by $c_i^{(3)}$ and $c_j^{(3)}$, and denote the results as $\widetilde{\mathrm{Cor}}_{<0,1,2>,3}$ and $\widetilde{\mathrm{Cor}}'_{<0,1,2>,3}$.
3) Check whether $\widetilde{\mathrm{Cor}}_{<0,1,2>,3}$ and $\widetilde{\mathrm{Cor}}'_{<0,1,2>,3}$ have two peaks meeting the following conditions: (1) Their values get a considerable increase. (2) Their time shifts are symmetric about the origin. If so, these two combinations $\boldsymbol{c_i}$ and $\boldsymbol{c_j}$ are identified as belonging to the same source.

The idea is also validated in our proof-of-concept experiment. We compute $\mathrm{Cor}_{<0,1,2>,3}$ and $\mathrm{Cor}'_{<0,1,2>,3}$ using $\boldsymbol{c_5}$ = (-5, -10, -16) and $\boldsymbol{c_4}$ = (-4, -7, -10) in Table 3 which belong to the same source, S_B. Fig. 9 plots the results and the original $\mathrm{Cor}_{0,3}$. In Fig. 9(a), two peaks located at sample -16 and -77 increase considerably. Similarly in Fig. 9(b), two peaks located at sample -10 and 51 also increase. We notice that the interval between two enhanced peaks in each subfigure is the same: 61 samples. Therefore, once shifting CCFs $\mathrm{Cor}_{<0,1,2>,3}$ and $\mathrm{Cor}'_{<0,1,2>,3}$ by $c_5^{(3)}$ (sample -16) and $c_4^{(3)}$ (sample -10) respectively, we can expect that the LOS-ECHO peak and the ECHO-LOS peak will be symmetric about the origin (sample -61 and 61), as shown in Fig. 9(c). By checking such symmetry, we can determine whether two combinations belong to the same source.

6.2 Discrimination between LOS and ECHO

Next, we identify the types of combinations: which one is LOS and which one is ECHO. Fig. 10 illustrates our observation. The blue and the red lines represent LOS and ECHO,

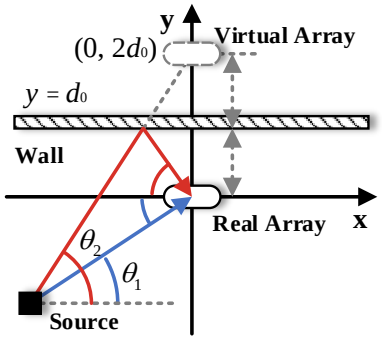


Fig. 10: $|\tan\theta_1|$ should be smaller than $|\tan\theta_2|$.

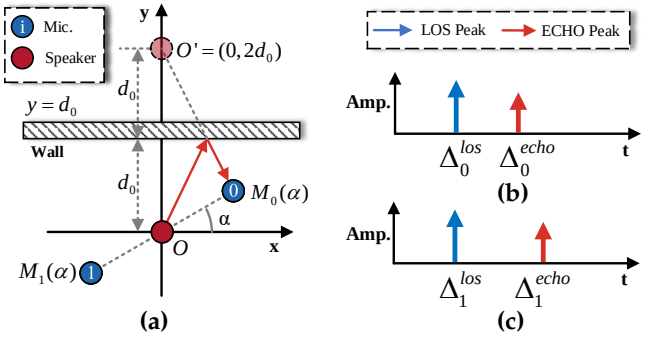


Fig. 11: The illustration of self-localization.

respectively. To make sure that these two lines intersect in Quadrant III or IV of this coordinate plane, the absolute value of the slope of the blue line should be smaller than that of the red line, namely $|\tan\theta_1| < |\tan\theta_2|$.

Based on this observation, we propose a simple but effective approach to distinguish LOS and ECHO: After recognizing two pure-peak combinations that belong to the same source and obtaining their DoAs, we compare $|\tan|$ of these two DoAs. The one with smaller $|\tan|$ is identified as LOS, and the other is ECHO.

7 Reverse Ray-Tracing

We localize sources via reverse ray-tracing. Again, we construct the coordinate system with the array at the origin and the nearby wall on the line $y = d_0$, as illustrated in Fig. 10. According to the plane mirror imaging principle, the points of the real array and the virtual array are symmetrical about the wall, and the virtual array is at point $(0, 2d_0)$. The two paths from the source to the real array and the virtual array can be formulated as:

$$\begin{cases} y = \tan(\theta_1 + \alpha)x, & \text{(LOS)} \\ y = \tan(\theta_2 - \alpha)x + 2d_0, & \text{(ECHO)} \end{cases} \tag{8}$$

where α is the array's orientation with respect to the wall. Therefore, the point of intersection of these two lines is the source location. Before reverse ray-tracing, the distance d_0 and the orientation α need to be calibrated. We cover it in the next section.

8 Self-Localization

To localize sources, we need to know two parameters: the distance d_0 between the array and the wall, and the array's orientation α with respect to the wall. Besides a microphone array, the smart speaker also has a speaker whose location is determined and known. This means that the smart speaker can localize itself based on the source location (i.e., its speaker) in return.

Specifically, we choose Frequency Modulated Continuous Waveform (FMCW) as our probing signal [20] to measure the distance d_0 and the orientation α. After transmitting FMCW, the smart speaker detects the arrival times of the FMCW by correlating the transmitted FMCW with the audio. For microphone M_i, as we explained in Section 4, there will be two dominant arriving paths, LOS and ECHO. Therefore, the smart speaker will detect two strong correlation peaks (LOS peak and ECHO peak), as illustrated in Fig. 11(b) and (c). Because the transmission of FMCW is also known, the smart speaker can calculate the propagation times that FMCW takes to arrive at M_i via the LOS and ECHO paths, i.e., Δ_i^{los} and Δ_i^{echo}.

As shown in Fig. 11(a), we construct the coordinate system by letting the speaker as the original and the nearby wall on the line $y = d_0$. Clearly, the length of the ECHO path from the speaker to M_i equals $||O^{'} M_i||_2$, where $O^{'}$ is the symmetric point of the speaker (O) about the wall, and $||\bullet||_2$ denotes the Euclidean norm. Therefore, we can build an equation for M_i:

$$||O^{'} M_i||_2 = v \times \Delta_i^{\text{echo}}. \tag{9}$$

Note that the coordinates of $O^{'}$ and M_i are only determined by the orientation α and the distance d_0, respectively. This means that there are only two unknowns (d_0 and α) in Eq. (9). Since each microphone can provide one constraint in the form of Eq. (9), as long as a smart speaker has more than one microphone, we can build a determined or overdetermined equation set to solve the distance and the orientation.

9 Localizability of *Symphony*

Here, the theoretical analysis and the numerical results of the localization capability are provided, which may provide us guidance on system deployment and adjustment.

9.1 Resolution and Coverage

9.1.1 Theoretical Analysis

In the following, we analyze how the resolution and the coverage are affected by three impacting factors: sampling rate F_s, array aperture $d_{\max}$, array-wall distance d_0.

Theorem 1 (Discrete Positions). *Symphony can only localize discrete positions, and the number of these localizable discrete positions equals* $2\left\lfloor \frac{d_{\max} F_s}{v} \right\rfloor^2$.

Proof. Reverse ray-tracing (Eq. (8)) is a function that maps the DoAs of LOS and ECHO, θ_1 and θ_2, to a point p in the room. We denote this function as $\mathcal{R}(\theta_1, \theta_2)$.

We define Θ as the set of all possible DoA values, i.e. $\theta_1 \in \Theta$ and $\theta_2 \in \Theta$. Due to limited spatial resolution, a DoA

θ can be mapped into at most $2\lfloor \frac{d_{\max}}{v} F_s \rfloor + 1$ discrete bins, where $d_{\max}$ is the maximum inter-microphone distance (i.e., the array aperture). This means $|\Theta| = 2\lfloor \frac{d_{\max}}{v} F_s \rfloor + 1$, where $|\bullet|$ denotes the size of a set. Because reverse ray-tracing $\mathcal{R}$ is a bijective function, the size of the *range* of $\mathcal{R}$ is equal to that of the *domain* of $\mathcal{R}$. Therefore,

$$|\text{range}(\mathcal{R})| = |\text{dom}(\mathcal{R})| \leq |\Theta \times \Theta| = \left(2\lfloor \frac{d_{\max}}{v} F_s \rfloor + 1\right)^2. \tag{10}$$

Note that the *range* of $\mathcal{R}$ is equivalent to the set of the locations that *Symphony* localizes. We define the locations yielded by reverse ray-tracing as the **discrete positions**. Eq. (10) shows that the number of the discrete positions is finite.

Moreover, the DoAs of LOS and ECHO are not arbitrary combinations of two DoA values in Θ, which means $\text{dom}(\mathcal{R}) \neq \Theta \times \Theta$. To meet the intersection criterion mentioned in Section 6.2, the $|\tan|$ value of θ_1 should be smaller than that of θ_2. Therefore, the *domain* of $\mathcal{R}$ is defined as

$$\text{dom}(\mathcal{R}) = \{(\theta_1, \theta_2) \mid \ |\tan\theta_1| < |\tan\theta_2|, \theta_1, \text{and } \theta_2 \in \Theta\}. \tag{11}$$

The size of the *domain* of $\mathcal{R}$ can be calculated as

$$|\text{dom}(\mathcal{R})| = \sum_{\theta_1 \in \Theta} \sum_{\theta_2 \in \Theta} \mathbf{1}\left[|\tan\theta_1| < |\tan\theta_2|\right] = \frac{(H-1)^2}{2}, \tag{12}$$

where $\mathbf{1}[\text{condition}]$ is a bool function that is equal to 1 if condition is true and 0 otherwise, and H denotes the size of Θ. Again, the number of the discrete positions is equal to $|\text{range}(\mathcal{R})|$, and thus to $|\text{dom}(\mathcal{R})|$. Substituting $H = 2\lfloor \frac{d_{\max}}{v} F_s \rfloor + 1$ into Eq. (12) leads to Theorem 1. □

Theorem 2 (Coverage of *Symphony*)**.** *The distribution of the discrete positions is scaled by the array-wall distance d_0.*

Proof. By solving Eq. (8), we can formulate the coordinate of the discrete position as follows:

$$\begin{cases} x = \dfrac{2}{\tan(\theta_2 - \alpha) - \tan(\theta_1 + \alpha)}\, d_0, \\ y = \dfrac{2\tan(\theta_1 + \alpha)}{\tan(\theta_2 - \alpha) - \tan(\theta_1 + \alpha)}\, d_0. \end{cases} \tag{13}$$

We can see that the coordinate is proportional to the array-wall distance d_0, i.e. $\text{norm}([x, \quad y]) \propto d_0$. This means that by changing the array-wall distance, we can scale up or down the coverage of the discrete positions. □

9.1.2 Distribution of Discrete Positions

Fig. 12 shows the discrete positions distributed in a 2D space, which actually reveals several interesting facts:

- Reverse ray-tracing localizes the source simply as its nearest discrete position. The localization error is mainly caused by the distance between the actual position of the source and its nearest discrete position.
- On average, the higher density of the discrete positions leads to a shorter distance between the source and its nearest discrete position. According to Theorem 1, if we increase the sampling rate F_s or the array aperture $d_{\max}$, the number and the density of discrete positions increase correspondingly, thus improving the localization resolution.
- The density of the discrete positions is distributed unevenly. A source close to the array can be localized with fine localization resolution, while a source far from the array implies coarse localization resolution.

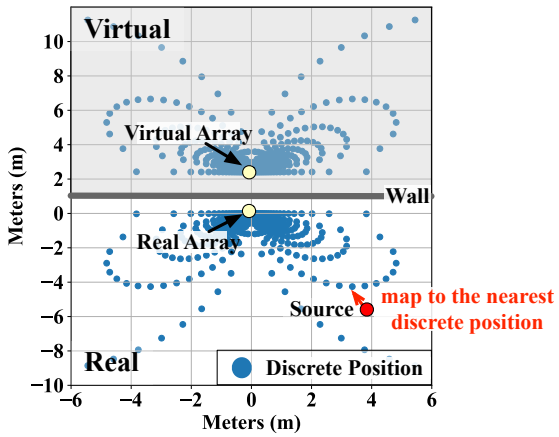


Fig. 12: The distribution of localizable discrete positions.

To further get an intuition of how they affect the distribution of the potential discrete positions, we compare the distributions attained by different settings of impacting factors in Fig. 13. We notice that

- As expected, the array-wall distance only changes the scale of the distribution. The greater the array-wall distance, the greater the region covered by the potential discrete positions, thus improving the coverage.
- On the other hand, the sampling rate changes not only the coverage, but also the number and the distribution shape of discrete positions. By increasing the sampling rate, we can increase both the coverage and the resolution of *Symphony*.

9.2 Discrimination between Multiple Sources

This subsection focuses on studying the ability to separate multiple sources. In fact, the problem of whether all sources can be separated and localized is equivalent to the problem of whether any two sources among these sources can be localized and separated. Therefore, we begin with a two-source case. We then study the general case where there are K sources ($K = 2, 3, 4...$), and present the numerical performance evaluated by Monte Carlo method.

9.2.1 Two-Source Case.

We first discuss the conditions on which *Symphony* fails to separate two sources, S_A and S_B[3].

To reverse ray-trace and localize S_A and S_B, two DoAs (LOS and ECHO) of both sources need to be calculated. However, if either DoA of S_A is very close to either DoA

3. In the following, we assume that each source is strong enough, so that their correlation peaks are not buried by noise.

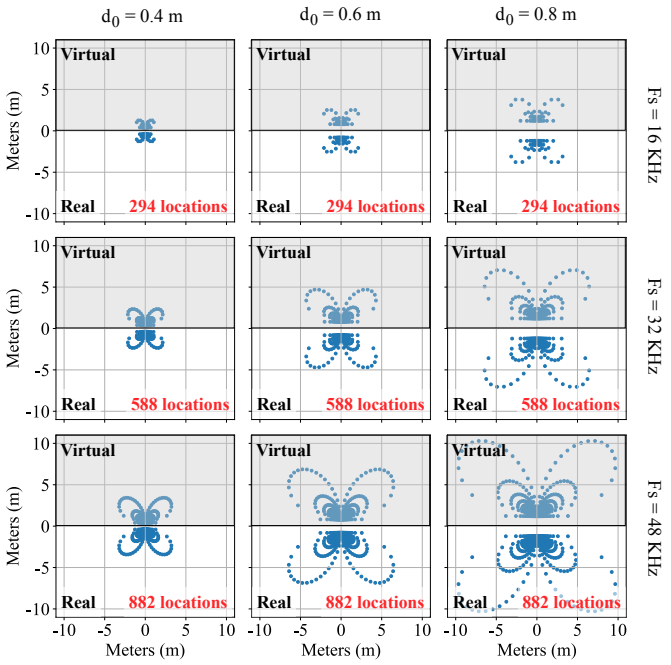


Fig. 13: The distributions of the discrete positions with different settings of the sampling rate F_s and the array-wall distance d_0.

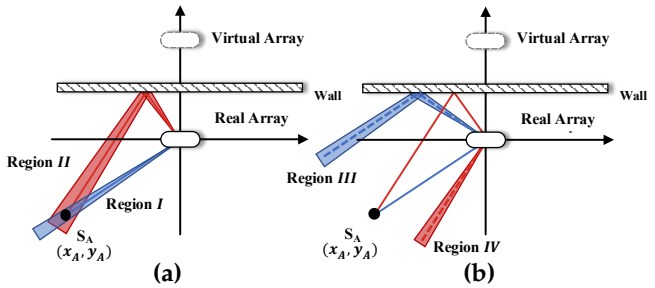


Fig. 14: When S_A is at (x_A, y_A), one of the DoAs of S_B will overlap with that of S_A if S_B is in Region I, II, III or IV.

of S_B, their corresponding pure peaks will be overlapped. Therefore, in this situation it is difficult to separate the DoAs of both sources.

Fig. 14 (a) and (b) illustrate this overlapping problem. Given the limited spatial resolution of the commercial array, if S_B is in region I, the DoAs of S_B's LOS and S_A's LOS will be too close to be separated. Similarly, if S_B is in region II, their DoAs of ECHO will also be overlapped. Further, if S_B is in region III or IV, the DoAs of S_A's LOS and S_B's ECHO will be merged, or the DoAs of S_A's ECHO and S_B's LOS will be merged, respectively..

Formally, let l_A and l_B denote the coordinates (i.e., locations) of S_A and S_B, and Region_l denotes the union of region I_l, II_l, III_l and IV_l (These four regions are determined by the source coordinate l). Therefore, the probability of successfully localizing both S_A and S_B is equal to

$$\mathbf{P}(S_A, S_B) = \int\int \mathbf{pdf}(l_A, l_B)\, \mathbf{1}\left[l_B \text{ not in Region}_{l_A}\right] \mathbf{d}l_A \mathbf{d}l_B, \quad (14)$$

where $\mathbf{pdf}(l_A, l_B)$ is the joint probability density function of l_A and l_B. If we further assume S_A and S_B are independently and uniformly distributed in the room, Eq. (14) can be rewritten as follows

$$\begin{aligned}
&\mathbf{P}(S_A, S_B) \\
&= \int \mathbf{pdf}(l_A) \left(\int \mathbf{pdf}(l_B)\, \mathbf{1}\left[l_B \text{ not in Region}_{l_A}\right] \mathbf{d}l_B \right) \mathbf{d}l_A \\
&= \int \mathbf{pdf}(l_A) \mathbf{P}(l_B \text{ not in Region}_{l_A}) \mathbf{d}l_A \\
&= \int \mathbf{pdf}(l_A) \left(1 - \frac{\text{area of Region}_{l_A}}{\text{area of the room}}\right) \mathbf{d}l_A. \quad (15)
\end{aligned}$$

Eq. (15) reveals a fact that the performance of separating and localizing multiple sources strongly depends on the spatial resolution of the array: When we increase the array aperture $d_{\max}$ or increase the sampling rate F_s, the area of Region_{l_A} will be reduced[4], and then the probability of localizing both S_A and S_B will increase.

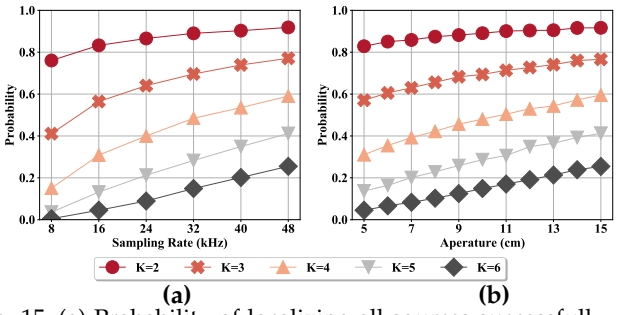


Fig. 15: (a) Probability of localizing all sources successfully v.s. Sampling Rate. ($d_{\max} = 15$ cm). (b) Probability of localizing all sources successfully v.s. Array Aperture. ($F_s = 48$ kHz)

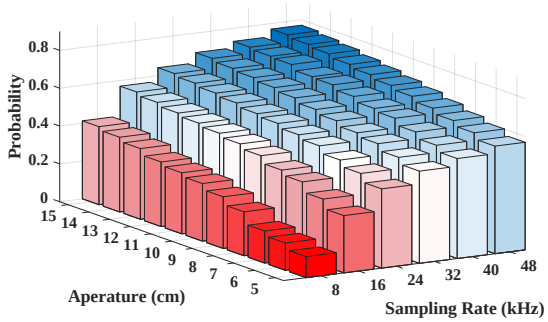


Fig. 16: Probability of localizing all sources successfully ($K = 3$) v.s. array aperture and sampling rate.

9.2.2 General Case

Here, we study the probability of successfully localizing all K sources, $S_1, S_2, \ldots S_K$. Similar to Eq. (14), the probability can be calculated as

$$\begin{aligned}
&\mathbf{P}(S_1, S_2, \ldots S_K) \\
&= \int_{\mathcal{L}} \mathbf{pdf}(\mathcal{L}) \mathbf{1}\left[\forall l_A, l_B \in \mathcal{L},\ l_B \text{ not in Region}_{l_A}\right] \mathbf{d}\mathcal{L}, \quad (16)
\end{aligned}$$

4. This is because an array with a higher spatial resolution allows to estimate a finer DoA, which means Region I, II, III and IV in Fig. 14 will be narrower.

where $\mathcal{L} = (l_1, l_2, \ldots, l_K)$. Unfortunately, Eq. (16) can not be computed analytically. To get an intuition that how $\mathbf{P}(S_1, S_2, \ldots S_K)$ varies with impacting factors, we use the Monte Carlo method to approximate $\mathbf{P}(S_1, S_2, \ldots S_K)$.

Specifically, for each setting of the sampling rate F_s, the aperture $d_{\max}$, and the number of source K, we conduct 10^5 trials to approximate the probability of successfully localizing all K sources, $\mathbf{P}(S_1, S_2, \ldots S_K)$. For each trial, we randomly generate K sources in a room (5m x 5m), and the distributions of K source locations are simulated independently and uniformly. The array-wall distance d_0 is set to 0.4 m.

Fig. 15(a) shows the impact of sampling rate on the success probability $\mathbf{P}(S_1, S_2, \ldots S_K)$ when the aperture is 15 cm. We vary the sampling rate from 8 kHz to 48 kHz (the maximum sampling rate that the most commercial arrays support), and then simulate and approximate the probability. As expected, the probability increases with the increase of the sampling rate. We also observe that $\mathbf{P}$ is largely affected by the number of sources K. In particular, when F_s = 48 kHz, the probability is 0.918 as $K = 2$, and significantly decreases to 0.255 as $K = 6$. However, it is worth noting that $\mathbf{P}(S_1, S_2, \ldots S_K)$ denotes the probability of the event that each source is localized successfully. Even if this event is false, we can still localize part of the sources, as long as these partial sources' Region_l have no overlap with each other and other sources' Region_l.

Fig. 15(b) also shows the impact of array aperture on $\mathbf{P}(S_1, S_2, \ldots S_K)$ when the sampling rate is 48 kHz. Similar to Fig. 15(a), the probability also increases with the increase of the aperture. Actually, by referring to Theorem 1 in Section 9.1, we can check that the spatial resolution is jointly determined by the sampling rate and the array aperture. Therefore, the impact of array aperture on the probability is similar to that of the sampling rate. This can also be verified in Fig. 16, which plots the joint impact of the array aperture and the sampling rate on the probability when $K = 3$.

10 Evaluation

10.1 Implementation

Hardware. As shown in Fig. 17, we built a prototype of *Symphony* using commercial off-the-shelf microphone arrays with two different layouts: Seeed Studio ReSpeaker 4-mic linear array [21] and ReSpeaker 6-mic circular array [22]. These two layouts are widely used in many popular smart speakers such as Amazon Echo and Alibaba Tmall Genie. The inter-distance of two adjacent microphones is 5 cm for both the linear array and the circular array. The speed of sound is assumed to be 343 m/s. The sampling rate is set to cover the whole audible frequency range, 48 kHz. Each array is sitting on top of a Raspberry Pi 4 Model B.

Software. We use the classical GCC-PHAT [16] method to calculate the CCF, which whitens the microphone signals to equally emphasize all frequencies. The computational efficiency of *Symphony* is bottlenecked by the calculation of CCFs. To accelerate the computation of GCC-PHAT, the Fast Fourier Transform (FFT) is used. Meanwhile, zero padding in the frequency domain and interpolation are also applied to reduce the discretization error after performing the FFT. To evaluate the computational efficiency of *Symphony* and prove that the localization overhead is affordable for resource-limited devices, we implement it in two computing platforms in Python: (1) A Raspberry Pi processing signals locally; and (2) a laptop processing samples streamed from the Raspberry Pi over WiFi.

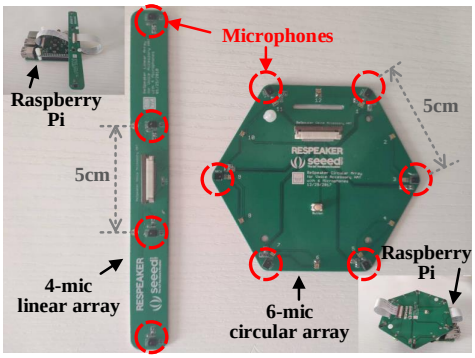


Fig. 17: 4-mic linear array and 6-mic circular array.

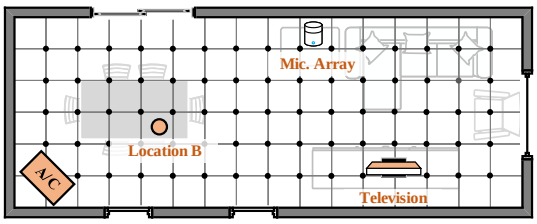


Fig. 18: The living room (8m x 3m).

10.2 Experimental Methodology

In the experiments, four volunteers (including both men and women) are recruited to record different voice commands. The durations of these voice commands are between 0.2s and 10s. We use a portable wireless speaker as the acoustic source, which plays the recorded voice commands at different positions in the room. The sound volume of the speaker is set to 60 dB SPL at 0.5 m.

We compare *Symphony* with VoLoc [12], a state-of-the-art approach for single-source localization using a single array. We implement VoLoc on both 6-mic and 4-mic arrays. The sampling rates of both VoLoc and *Symphony* are set to 48 kHz.

We conduct multiple experiments to evaluate the performance of *Symphony*. In the following subsections, we first present the localization result in the multi-source scenario (Section 10.3). Then, we compare the performance of Symphony with that of VoLoc in the single-source scenario, under both clean and noisy conditions (Section 10.4). Next, we evaluate the impact of DoA overlapping on localization (Section 10.5). In addition, we examine other performance parameters,, including DoA estimation accuracy (Section 10.6) and computational efficiency (Section 10.7).

10.3 Localization in the Multi-Source Scenario

We first evaluate *Symphony*'s localization accuracy in the multi-source scenario. We conduct the experiment in a living room of 8m x 3m (Fig. 18). The reverberation time T60 (the time the sound takes to decay by 60 dB) of the room is about

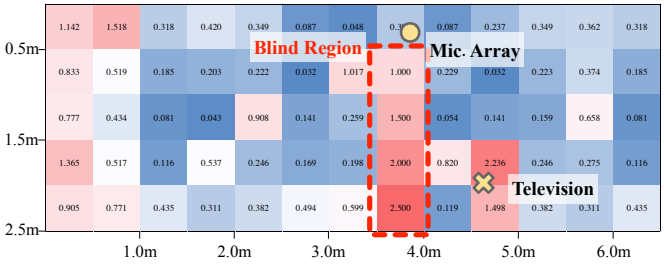


Fig. 19: Heatmap of *Symphony*'s localization error.

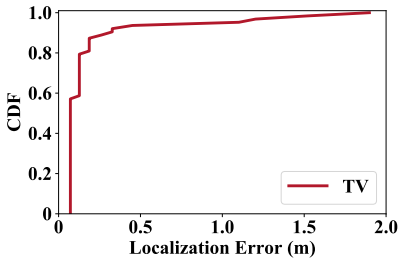


Fig. 20: Localization error of the television.

0.55 s. A 4-mic array is placed 0.35m away from the wall. The television is powered on and kept making sound (65 dB SPL at 0.5 m), while the air conditioner is powered off. We use the portable speaker as the source and deploy it at different positions in the room, denoted by the black dots in Fig. 18. So we have two sources in this experiment. Fig. 19 shows the localization error of the speaker, while Fig. 20 plots the localization error of the television. We can see that:

- *Symphony*'s localization error tends to increase as the distance between the array and the source increases. According to the aforementioned far-field effect, due to the limited resolution of the hardware, the minimum measurable change in DoA is fixed.
- There is a blind region of localization, as shown in Fig. 19: When the connected line between the source and the array is perpendicular to the wall, *Symphony*'s localization performance is quite poor. To better understand this, refer to Fig. 10. The perpendicular case in Fig. 19 is equivalent to the case where the real array, the virtual array, and the source are collinear. In this case, the slope of LOS is equal to that of ECHO, i.e., $\theta_1 = \theta_2$. It is impossible to know this source location using reverse ray-tracing. In other words, any source close to the negative Y-axis cannot be localized. In our implementation, *Symphony* simply outputs the array location when detecting $\theta_1 \approx \theta_2$.
- The localization performance degrades when two sources are close. This is because when the sources are close, the DoAs of their LOS and ECHO paths become also close and indistinguishable, thus confusing *Symphony* during the DoA estimation.

10.4 Localization in the Single-Source Scenario

We compare the localization accuracy of *Symphony* with VoLoc in the single-source scenario, under both clean and noisy conditions. Due to the ability to distinguish multiple acoustic sources, we expect that *Symphony* has better performance than VoLoc in noisy environments. The experiments are conducted in the living room depicted in Fig. 18. We play voice commands via a portable speaker at different positions.

Localization under clean condition. We place the 4-mic array 0.4m away from the wall. We conduct experiments at night, when it is very quiet in the living room. The volume of background noise is lower than 20 dB SPL. Fig. 21 shows the localization errors of *Symphony* and VoLoc. The median error of VoLoc is 0.314 m, which is slightly better than that of *Symphony*, 0.387 m. The slight gap in performance is because VoLoc uses a fine-grained but exhausting searching method to localize the source, which produces more accurate results in the ideal case.

Localization under noisy condition. We place the 6-mic array 0.4 m away from the wall. We conduct the experiment during the day. The volume of background noise is around 40-46 dB SPL. In addition, the air conditioner is kept on, producing noise at about 38-42 dB SPL (measured at the array position). The television is powered off. For completeness of the result, we also plot the localization result of *Symphony* and VoLoc using the 6-mic array under clean condition[5]. In the noisy environment, *Symphony* only suffers slight performance degradation: the median error increases from 0.578 m to 0.662 m. The performance of VoLoc suffers significant degradation: the median error increases from 0.536 m to 0.937 m. VoLoc is susceptible to noise because it only uses a short time window of the received signals to search for DoAs. So its performance is likely to be affected by low signal-to-noise ratio. *Symphony* uses the whole voice signals to obtain the CCFs, which is robust against the noise.

Moreover, by comparing the results under the clean condition in Fig. 21 and Fig. 22, we find that for both *Symphony* and VoLoc, the 4-mic array version performs better than the 6-mic array version. This is because the 4-mic array has a larger aperture than the 6-mic array, leading to a finer spatial resolution [6].

10.5 Impact of DoA Overlapping

As introduced in Section 9.2, when one source is in Region I, II, III, or IV of the other source, *Symphony* might not separate them due to DoA overlapping. Next, we study the impact of DoA overlapping on localization.

This experiment involves three sources (Source A, B, and C). We use the TV and a portable speaker as S_A and S_B, respectively. S_A and S_B are kept speaking at two fixed locations (i.e., the location of TV and the location B in Fig. 18) in the living room. We then use another portable speaker as S_C to speak at different locations across this room (the black dots in Fig. 18). We place the 4-mic array 0.4 m away from the wall to localize these sources. Fig. 23 shows the localization error of these three sources separately. The median errors of S_A, S_B, and S_C are 0.337 m, 0.155 m and 0.710 m, respectively. Apparently, the localization result of S_C is much worse than those of S_A and S_B. One main reason is the following: In the experiment, S_A and S_B are at two fixed and well separated locations. This ensures that the DoAs of these two sources are always separable by the array. On the other hand, S_C is placed at different locations. It is more likely for the DoAs of S_C to overlap

5. The experiments are conducted at the night with the air-conditioner and the television turned off

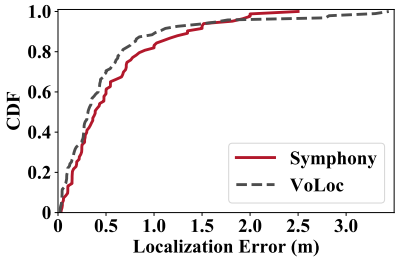


Fig. 21: Localization error on ideal conditions (4-mic).

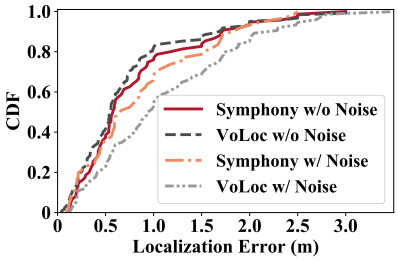


Fig. 22: Localization error w/ or w/o noise (6-mic).

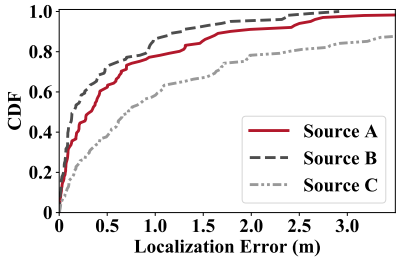


Fig. 23: Localization error of three sources.

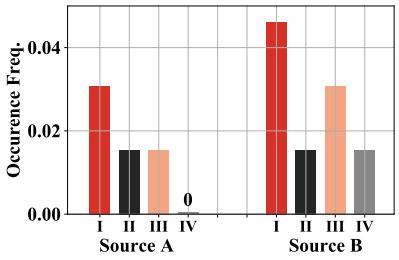


Fig. 24: The frequency of overlapping.

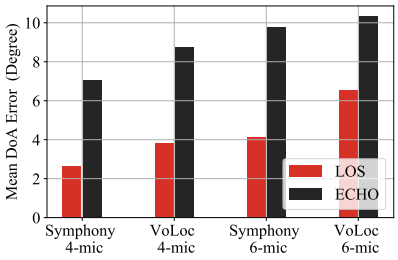


Fig. 25: DoA estimation error.

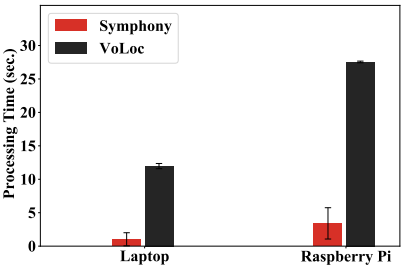


Fig. 26: Processing time.

with the DoAs of other sources. Furthermore, we display the occurrence frequency of the event that the measured DoAs of S_C overlap with those of S_A and S_B in Fig. 24 in terms of the four overlapping types (see Section 9.2). The frequencies of DoA overlapping between S_A and S_C, and between S_B and S_C are 6.2% and 10.8%, respectively. This partially explains why the result of S_A is slightly better than that of S_B.

10.6 DoA Estimation Accuracy

Here, we evaluate the accuracy of DoA estimation, which directly determines the accuracy of localization. Fig. 25 compares *Symphony* and the baseline on DoA estimation error in the environment with ambient noise. As we can see, the DoA estimation of the LOS path, as expected, is more accurate than that of the ECHO path. We also observe that the 4-mic array attains a finer DoA resolution than the 6-mic array. As before, this is because the 4-mic array has a larger aperture than the 6-mic array. Meanwhile, in the noisy environment, *Symphony* achieves a better DoA estimation than the baseline, which is consistent with the results shown in Fig. 22.

10.7 Computational Efficiency

We compare the average processing time of *Symphony* and VoLoc. The processing time refers to the time these methods takes to output source locations after receiving the audio. Fig. 26 shows the results. The average time consumption of *Symphony* for the laptop and the Raspberry Pi are 1.4 seconds and 3.4 seconds, respectively. *Symphony* significantly reduces the processing time, being 9x and 8x faster than the baseline on the laptop and the Raspberry Pi, respectively. This is because VoLoc models the localization problem as a highly non-convex optimization problem, and resorts to the brute-force searching method to localize the source. On the contrary, *Symphony* leverages the sparsity of peaks to avoid sample-wise searching, and leverages the array geometry to quickly narrow down to pure-peak combinations.

11 Related Work

Localization by Exploiting Room Information. The works closer to ours are VoLoc [12] and MAVL [23]. Both VoLoc and MAVL exploit the wall reflection to localize an acoustic source. VoLoc proposes a novel iterative align-and-cancel algorithm for multipath DoA estimation. Further, MAVL [23] improves the estimation accuracy of DoA by leveraging multi-resolution analysis in the time-frequency domain. PACE [24] is another work close to *Symphony*. PACE achieves localization by exploiting the information from both structure-borne and air-borne sounds for range estimation and DoA estimation, respectively. These three works focus on localizing a single source and thus might be vulnerable to acoustic interference in practice, while *Symphony* supports localizing multiple sources concurrently. In [25], not only the nearby wall, the whole room information is measured by a Kinect depth sensor, and then used to localize a clapping sound. [26], [27] measure the shape of the room and localize the source by transmitting and receiving known signals. iLocScan [28] achieves source localization and space scanning simultaneously by fully exploiting the information embedded in the signal multipath.

Concurrent Localization. The Global Positioning System (GPS) supports billions of clients localizing themselves concurrently. Such nearly unlimited scalability is partially because channels are utilized only by satellites. This design is rather appealing. Chorus [9] introduces a different scheme in which ultra-wideband (UWB) tags do not transmit at all and localize themselves only by measuring Channel Impulse Responses (CIRs), which significantly improves scalability compared with standard UWB two-way ranging. However, when clients actively join the procedures of localization, the scalability decreases dramatically, because clients should transmit in different time slots to avoid collisions. Many methods are designed to tolerate collisions from multiple sources, paving the way for concurrent localization [29], [30], [31], [32], [33]. Inspired by these works, we

measure the DoAs from the collided signals.

Localization with Multiple Arrays or Anchors. Distributed microphone / antenna arrays have been used to localize various sources, including smartphones [34], [35], WiFi clients [5], RFID tags [6], birds [7], and bumblebees [8].

DoA Estimation. *Symphony* uses a popular method, GCC-PHAT [16], to compute CCF. In fact, *Symphony* might be extended to other DoA estimation algorithms like Multiple Signal Classification (MUSIC [36]) and Estimation of Signal Parameters via Rotational Invariance Techniques (ESPRIT [37]), because the underlying problem is the same: the ambiguity of peaks, no matter they are from the CCF (GCC-PHAT), or from the pseudo-spectrum (MUSIC and ESPRIT).

12 Discussion

- **3D Localization.** *Symphony* virtually contains two microphone arrays, the real one and the virtual one, and thus only supports localization in 2D.
- **Non-Line-of Sight.** A source is localized as the intersection of the LOS path and the ECHO path. When the LOS path is blocked, *Symphony* will fail to localize the source.
- **The Array-Wall Distance.** If the array is far away from the wall, some problems will arise: (1) ECHO would experience more attenuation and its strength would not be comparable to LOS. (2) ECHO may not be the second arriving path: the path reflected by the wall close to the source may be shorter than ECHO. On the other hand, if the array is too close to the wall, *Symphony* will also perform poorly because the real array is close to the virtual array, and the far-field effect happens again.
- **Number of Microphones.** *Symphony* requires at least 3 microphones to exploit geometric redundancy. For smart speakers with only two microphones (e.g., Google Home), *Symphony* might not be applied.
- **Moving Objects.** *Symphony* assumes that sources are static when sources are active. Localizing moving objects will be another problem, which is out of our scope.
- **Number of Sources.** *Symphony* does not explicitly guarantee that a certain number of sources must be localized, it just provides a "best-effort" service. The whole localization procedure does not regulate that a certain number of sources must be found by *Symphony*. Instead, *Symphony* finds as many pure-peak combinations as possible, and thus localizes as many sources as possible. This implicitly determines the number of sources.
- **Identification of Sources** *Symphony* enables an array only to localize the sources, but not to identify the sources (i.e., it does not determine which device or person each source corresponds to).

13 Conclusion

We demonstrate the feasibility of using a single microphone array to localize multiple acoustic sources concurrently. We believe *Symphony* will enable new applications for location-aware services. In our design, we passively exploit the ECHO path to tackle the problem of far-field effect. We may further develop this idea by actively customizing the surroundings of the array, thus introducing more predictable multi-paths. Based on these, we may be able to extract more spatial information and thus localize the source more precisely.

References

[1] G. Zhang, C. Yan, X. Ji, T. Zhang, T. Zhang, and W. Xu, "Dolphinattack: Inaudible voice commands," in *Proceedings of ACM Conference on Computer and Communications Security*, 2017.

[2] D. Kumar, R. Paccagnella, P. Murley, E. Hennenfent, J. Mason, A. Bates, and M. Bailey, "Skill squatting attacks on Amazon Alexa," in *Proceedings of USENIX Security Symposium*, 2018.

[3] N. Roy, H. Hassanieh, and R. R. Choudhury, "Backdoor: Making microphones hear inaudible sounds," in *Proceedings of ACM MobiSys*, 2017.

[4] T. Sugawara, B. Cyr, S. Rampazzi, D. Genkin, and K. Fu, "Light commands: Laser-based audio injection attacks on voice-controllable systems," in *Proceedings of USENIX Security Symposium*, 2020.

[5] J. Xiong and K. Jamieson, "Arraytrack: A fine-grained indoor location system," in *Proceedings of USENIX NSDI*, 2013.

[6] J. Wang, D. Vasisht, and D. Katabi, "RF-IDraw: Virtual touch screen in the air using RF signals," 2014.

[7] T. C. Colliera, A. N. G. Kirschel, and C. E. Taylor, "Acoustic localization of antbirds in a mexican rainforest using a wireless sensor network," *The Journal of the Acoustical Society of America*, vol. 128, no. 1, pp. 182–189, 2010.

[8] V. Iyer, R. Nandakumar, A. Wang, S. B. Fuller, and S. Gollakota, "Living IoT: A flying wireless platform on live insects," in *Proceedings of ACM MobiCom*, 2019.

[9] P. Corbalán, G. P. Picco, and S. Palipana, "Chorus: UWB concurrent transmissions for GPS-like passive localization of countless targets," in *Proceedings of ACM/IEEE IPSN*, 2019.

[10] I. . W. Group, "IEEE standard for local and metropolitan area networks—part 15.4: Low-rate wireless personal area networks (LR-WPANs)," *IEEE STD*, vol. 802, pp. 4–2011, 2011.

[11] Q. Lin, Z. An, and L. Yang, "Rebooting ultrasonic positioning systems for ultrasound-incapable smart devices," in *Proceedings of ACM MobiCom*, 2019.

[12] S. Shen, D. Chen, Y. Wei, Z. Yang, and R. R. Choudhury, "Voice localization using nearby wall reflections," in *Proceedings of ACM MobiCom*, 2020.

[13] Govee, "Govee 32.8ft LED strip lights works with Alexa Google Home," https://www.amazon.com/Govee-Wireless-Control-Kitchen-Million/dp/B07WHP2V77/, 2020, accessed: 2020-10-02.

[14] BlissLights, "home theater lighting," https://www.amazon.com/BlissLights-Sky-Lite-Projector-Bedroom/dp/B084DCF429/, 2020.

[15] J. C. Curlander and R. N. McDonough, *Synthetic aperture radar*. Wiley, New York, 1991.

[16] C. Knapp and G. Carter, "The generalized correlation method for estimation of time delay," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 24, no. 4, pp. 320–327, 1976.

[17] J. Benesty, J. Chen, and Y. Huang, *Microphone array signal processing*. Springer Science & Business Media, 2008, vol. 1.

[18] W. Wang, J. Li, Y. He, and Y. Liu, "Symphony: localizing multiple acoustic sources with a single microphone array," in *Proceedings of ACM SenSys*, 2020.

[19] S. M. Goldfeld, S. M, and R. E. Quandt, *Nonlinear methods in econometrics*. North-Holland Pub. Co., 1972.

[20] W. Mao, J. He, and L. Qiu, "Cat: high-precision acoustic motion tracking," in *Proceedings of ACM MobiCom*, 2016, pp. 69–81.

[21] Seeed, "Respeaker 4-mic linear array kit for Raspberry Pi," https://wiki.seeedstudio.com/ReSpeaker_4-Mic_Linear_Array_Kit_for_Raspberry_Pi/, 2020.

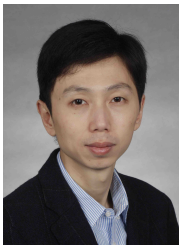
[22] ——, "Respeaker 6-mic circular array kit," https://wiki.seeedstudio.com/ReSpeaker_6-Mic_Circular_Array_kit_for_Raspberry_Pi/, 2020.

[23] M. Wang, W. Sun, and L. Qiu, "MAVL: multiresolution analysis of voice localization," in *USENIX NSDI, April 12-14, 2021*, 2021.

[24] C. Cai, H. Pu, P. Wang, Z. Chen, and J. Luo, "We hear your PACE: passive acoustic localization of multiple walking persons," *ACM IMWUT*, vol. 5, no. 2, pp. 55:1–55:24, 2021.

[25] I. An, M. Son, D. Manocha, and S. Yoon, "Reflection-aware sound source localization," in *Proceedings of IEEE International Conference on Robotics and Automation*, 2018.

[26] I. Dokmanić, R. Parhizkar, A. Walther, Y. M. Lu, and M. Vetterli, "Acoustic echoes reveal room shape," *National Academy of Sciences*, vol. 110, no. 30, pp. 12 186–12 191, 2013.
[27] M. Krekovic, I. Dokmanic, and M. Vetterli, "EchoSLAM: Simultaneous localization and mapping with acoustic echoes," in *Proceedings of IEEE ICASSP*, 2016.
[28] C. Zhang, F. Li, J. Luo, and Y. He, "iLocScan: harnessing multipath for simultaneous indoor source localization and space scanning," in *Proceedings of ACM SenSys, Memphis, Tennessee, USA, November 3-6, 2014*, 2014, pp. 91–104.
[29] P. Hu, P. Zhang, and D. Ganesan, "Laissez-faire: Fully asymmetric backscatter communication," in *Proceedings of ACM SIGCOMM*, 2015.
[30] J. Ou, M. Li, and Y. Zheng, "Come and be served: parallel decoding for COTS RFID tags," in *Proceedings of ACM MobiCom*, 2015.
[31] M. Jin, Y. He, X. Meng, Y. Zheng, D. Fang, and X. Chen, "Fliptracer: Practical parallel decoding for backscatter communication," in *Proceedings of ACM MobiCom*, 2017.
[32] M. Jin, Y. He, X. Meng, D. Fang, and X. Chen, "Parallel backscatter in the wild: When burstiness and randomness play with you," in *Proceedings of ACM MobiCom*, 2018.
[33] M. Jin, Y. He, C. Jiang, and Y. Liu, "Fireworks: Channel estimation of parallel backscattered signals," in *Proceedings of ACM/IEEE IPSN*, 2020.
[34] D. B. Haddad, W. A. Martins, M. d. V. Da Costa, L. W. Biscainho, L. O. Nunes, and B. Lee, "Robust acoustic self-localization of mobile devices," *IEEE Transactions on Mobile Computing*, vol. 15, no. 4, pp. 982–995, 2015.
[35] K. Liu, X. Liu, and X. Li, "Guoguo: Enabling fine-grained smartphone localization via acoustic anchors," *IEEE Transactions on Mobile Computing*, vol. 15, no. 5, pp. 1144–1156, 2015.
[36] R. Schmidt, "Multiple emitter location and signal parameter estimation," *IEEE Transactions on Antennas and Propagation*, vol. 34, no. 3, pp. 276–280, 1986.
[37] R. H. R. III and T. Kailath, "Esprit-estimation of signal parameters via rotational invariance techniques," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 37, no. 7, pp. 984–995, 1989.



Yuan He is an associate professor in the School of Software and BNRist of Tsinghua University. He received his B.E. degree in the University of Science and Technology of China, his M.E. degree in the Institute of Software, Chinese Academy of Sciences, and his PhD degree in Hong Kong University of Science and Technology. His research interests include wireless networks, Internet of Things, pervasive and mobile computing. He is a member of IEEE and ACM.

Weiguo Wang is currently a PhD. student in Tsinghua University. He received his B.E. degree in the University of Electronic Science and Technology of China (UESTC). His research interests include wireless sensing and communication.

Yunhao Liu received his B.S. degree in Automation Department from Tsinghua University, and an M.A. degree in Beijing Foreign Studies University, China. He received an M.S. and a Ph.D. degree in Computer Science and Engineering in Michigan State University, USA. He is now a professor at Automation Department and Dean of the GIX in Tsinghua University, China. He is a Fellow of IEEE and ACM.

Jinming Li is currently a graduate student in Tsinghua University. He received his B.E. degree in Tsinghua University. His research interests include wireless networks and Internet of Things.