Toward Reliable Localization by Unequal AoA Tracking

Tzu-Chun Tai
National Chiao Tung University
HsinChu, Taiwan
taitc@cs.nctu.edu.tw

Kate Ching-Ju Lin
National Chiao Tung University
HsinChu, Taiwan
katelin@cs.nctu.edu.tw

Yu-Chee Tseng
National Chiao Tung University
HsinChu, Taiwan
yctseng@cs.nctu.edu.tw

Abstract

Emerging applications require the location information of clients to enable human-environment interactions or personalized services. With an increasing number of antennas equipped in today's wireless devices, recent research has shown the possibility of sub-meter level localization based only on the angle of arrival (AoA) of WiFi signals. While most existing work provides promising median accuracy, tail performance is usually far worse. We observe from measurements that the root cause is *unequal AoA estimation reliability*. In some critical areas, a small variation in the channel state information of signals could introduce an extremely large AoA estimation error. With this observation, we propose UAT (Unequal Angle Tracking), a confidence-aware AoA-based localization system. We show that unequal reliability of AoA measures can be mathematically quantified, allowing a system to weigh the estimates of different APs according to their confidence. Our testbed evaluation shows that UAT's confidence-aware design provides reliable decimeter level localization for around 90% of locations. UAT is especially effective for unreliable areas and can reduce their localization errors by 27.5%, as compared to reliability-oblivious designs.

CCS Concepts

• **Networks** *Location based services*; • **Hardware** *Signal processing systems*.

Keywords

AoA Estimation; Localization; Unequal Tracking

ACM Reference Format:
Tzu-Chun Tai, Kate Ching-Ju Lin, and Yu-Chee Tseng. 2019. Toward Reliable Localization by Unequal AoA Tracking. In *The 17th Annual International Conference on Mobile Systems, Applications, and Services (MobiSys '19), June 17–21, 2019, Seoul, Republic of Korea.* ACM, New York, NY, USA, 13 pages. https://doi.org/10.1145/3307334.3326103

1 Introduction

The demand for accurate indoor localization has been rapidly increasing. More and more applications now rely on the location information to provide location-aware services. For example, smart homes, smart factories, augmented reality (AR) and virtual reality (VR) need the location information of a client to enable human-environment interactions or control IoT devices adaptively. Millimeter-wave (mmWave) wireless networks leverage narrow beams to support multi-gigabit data rates. With the user location information, the complexity of beam searching can be significantly reduced. To serve those emerging interactive applications, a localization system that achieves decimeter-level accuracy becomes important and necessary nowadays.

While today's devices support an increasing number of antennas, several recent systems [14, 31, 43] have demonstrated the potential of leveraging antenna arrays to enable sub-meter level localization based on the angle of arrival (AoA) of wireless signals. Unlike traditional RSSI-based approaches [5, 28, 46], which typically suffer from signal strength fluctuation, AoA-based systems exploit the spatial dimension of multiple antennas to extract only the phase of channel state information (CSI) and, thereby, can be free from being disturbed by dynamic channel fading.

While existing solutions have shown promising results, those efforts usually focus on the median accuracy. However, in many of today's systems, the 90th percentile of the accuracy can be about 5× worse than the median [14, 43], hindering those systems from being reliably used in practice. We found that the root cause of this large tail error is *unequal AoA estimation reliability*. Our empirical measurements show that AoA estimation can be accurate in most locations, but has a considerable error (up to 135 degrees) in some dead zones. Not only this, the variation of AoA estimation is especially high when a transmitting client is not static, making AoA estimation highly uncertain. As most of the existing AoA-based approaches collect the estimation of multiple APs to determine the target location, any AP with a large AoA error could bias the localization result. The goal of this work is hence to quantify the AoA estimation reliability and develop a system that can avoid being disturbed by unreliable AoA measures, thereby shrinking the localization performance gap between the median and the tail.

In this paper, we present UAT (*Unequal AoA Tracking*), a system that enables multiple access points (AP) to perform *adaptive confidence-aware* AoA-based localization. While most localization algorithms aims at improving the accuracy of AoA measurements, our goal rather is to quantify the confidence level of those AoA measurements. Hence, UAT can be integrated with any AoA estimation algorithm, e.g., ArrayTrack [43] or SpotFi [14], to enhance their reliability. UAT is designed based on a key observation that, while a client may be located in a unreliable area of one AP, it is very unlikely that its location happens to be within the dead zones of all the APs. Therefore, if we can adaptively prioritize the APs based on their confidence levels, we should be able to rule out unreliable measurements. By doing this, UAT leverages the diversity

MobiSys '19, June 17–21, 2019, Seoul, Republic of Korea

ACM ISBN 978-1-4503-6661-8/19/06…$15.00
https://doi.org/10.1145/3307334.3326103

of multiple APs to cope with unequal AoA errors and, hence, enhance localization reliability.

To realize UAT in practice, however, we should answer a key question: how can an AP determine the confidence level of an AoA measure? To do so, we first identify the relationship between an AoA and the phases of its corresponding CSIs received by multiple antennas. We observe that, at a very high level, a constant error in the CSI phase could lead to a very different error in AoA, as a result making AoA estimation unequally reliable in different areas. Besides, another source of AoA errors comes from channel dynamics. Even slight movement of a client device could make AoA measures highly variant. The mixture of the two error sources, i.e., location-dependent errors and random dynamics, make it hard to infer AoA uncertainty.

Since it is not easy to directly learn the angle error distribution from the mixed error sources, we develop a two-phase confidence analysis to infer AoA reliability. We notice that AoA errors caused by channel variation is a random variable independent of client locations. However, location-dependent errors introduced by phase-to-angle conversion are actually deterministic and can be derived mathematically. To eliminate the impact of user locations, we hence propose to instead learn the error distribution in the phase domain and then transform such a phase-domain error distribution to the angle domain with consideration of phase-to-angle bias. By such non-linear error transformation, UAT better captures the real angle error distribution and determines the confidence level of an AP. Then, UAT leverages the inferred reliability of multiple APs to perform confidence-aware localization. With such adaptation, UAT prioritizes AoA measures based on their unequal errors so as to improve localization reliability.

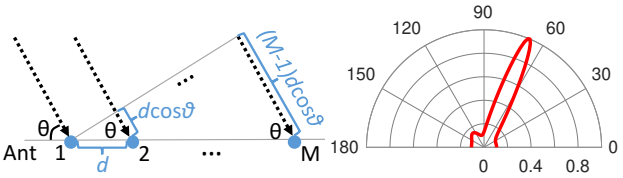
Our work has the following contributions:

- We identify the issue of unequal AoA reliability and conduct measurement studies to support our observations.
- We mathematically derive location-dependent AoA errors and infer AoA confidence by a two-phase error distribution analysis.
- With the estimated AoA confidence, we propose UAT, a confidence-aware localization system, which detects unreliable measures and leverages the diversity of APs to achieve reliable localization.
- Our experimental results show that UAT reliably provides decimeter-level localization for most locations. For challenging locations in the dead zones of some risky APs, UAT reduces the error distance by 27.5% as compared to equal AoA sampling.

As many recent advances [38, 42] have further fused other features, e.g., RSSI, ToF and TDoF, to improve localization accuracy, we believe that our effort on enhancing AoA reliability can be incorporated with those fusion systems to improve their tail performance.

2 Background and Motivation

We start by introducing the theory of the MUSIC algorithm [30], one of the most well-known AoA estimation algorithms. We then explain the potential causes of unequal AoA reliability alone with empirical measurements that motivate this work.



(a) Example of LOS AoA (b) Example of AoA Spectrum

Figure 1: AoA Estimation. (a) A signal traverses through parallel paths of different lengths, resulting in phase differences at antennas. (b) AoA spectrum can be learned by MUSIC.

2.1 Primer of AoA Estimation

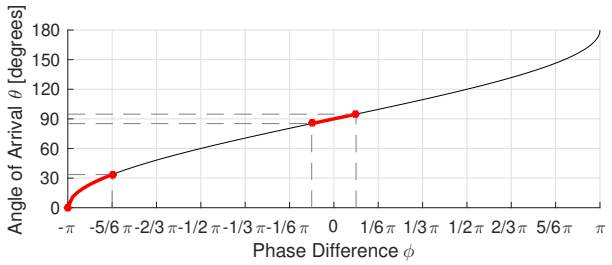
When a wireless signal X sent by a client traverses through space, the signal received by a receiving antenna can be expressed as $Y_k = H_k X + N_k$, where H_k is the Channel State Information (CSI) between the transmitter and the receiver over OFDM subcarrier k and N_k is the Gaussian white noise. The CSI H_k is determined by both signal fading and propagation delay over multiple paths the signal traverses through. If the signal only goes through a single line-of-sight (LOS) path, then, $H_k = \alpha e^{-2j\pi f_k t}$, where α is the received signal amplitude, f_k is the carrier frequency of subcarrier k and t is the traversing time along the LOS path.

The angle of an arriving signal can typically be measured by a uniform linear antenna array based on time difference of arrival. Say an antenna array is equipped with M antennas, and any two adjacent antennas are separated by a space d. Assume that the distance between a transmitter and the receiving antenna array is much longer than the antenna space d. Then, the signals from the transmitter to different receiving antennas can be deemed as traversing along parallel paths. While the signal coming from a particular angle θ traverses through those parallel paths of different lengths, they will arrive at different antennas in slightly different times. In particular, as shown in Fig. 1(a), the distance of a path arriving from angle θ to the first antenna is slightly shorter than that to the second antenna. The distance difference equals $d\cos\theta$, which corresponds to an additional signal traversing time of $d\cos\theta/c$, where c is the light speed. Such time differences introduce an additional phase (relative to the first antenna) to each of the other antennas. As the wavelength $\lambda_k = c/f_k$, the array steering vector $\mathbf{a}_k(\theta)$ can be characterized as a function of the signal's incoming angle as follows:

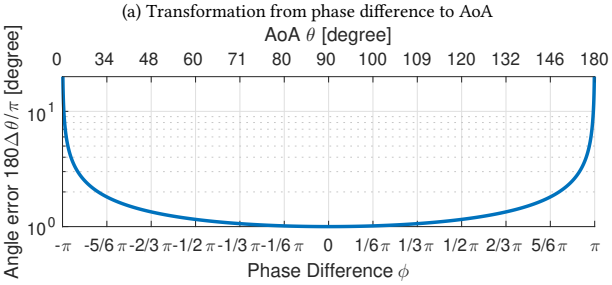
$$\mathbf{a}_k(\theta) = e^{-2j\pi f_k t} \begin{bmatrix} 1 \\ e^{-2j\pi d\cos\theta/\lambda_k} \\ e^{-2j\pi 2d\cos\theta/\lambda_k} \\ \vdots \\ e^{-2j\pi(M-1)d\cos\theta/\lambda_k} \end{bmatrix}. \quad (1)$$

In a multipath environment, a signal would traverse through multiple paths, including the LOS path and other reflected paths. The signal received by an antenna is hence the linear superposition of the signals from all the paths, i.e.,

$$Y_k = [\mathbf{a}_k(\theta_1)\ \mathbf{a}_k(\theta_2)\ \cdots\ \mathbf{a}_k(\theta_L)] \begin{bmatrix} X_1 \\ X_2 \\ \vdots \\ X_L \end{bmatrix} + N_k, \quad (2)$$



(a) Transformation from phase difference to AoA



(b) AoA error caused by a constant phase error

Figure 2: Unequal AoA errors. (a) The phase-to-angle mapping function, in which a fixed phase error leads to unequal angle errors, as the two red segments illustrated. (b) The derivative of the phase-to-angle function quantifies unequal angle errors caused by a small phase error (in degrees).

where L is the number of paths and $\mathbf{a}_k(\theta_l)$ is the steering vector of path l from angle θ_l over subcarrier k. By equipping with multiple receiving antennas, we can leverage their spatial dimensions to separate the signals and identify their incoming angles $\theta_l, l = 1, \cdots, L$. Given H_k across all the subcarriers k, MUSIC [30] constructs the AoA spectrum of a received signal, as illustrated in Fig. 1(b). Each point in the spectrum represents the likelihood of an incoming angle. As the LOS path usually results in the strongest receiving power, the *peak AoA* (the angle with the maximal amplitude in the AoA spectrum) is most likely the LOS AoA. Existing AoA-based localization algorithms then leverage the LOS AoAs of multiple APs to localize a target client.

2.2 Unequal AoA Error

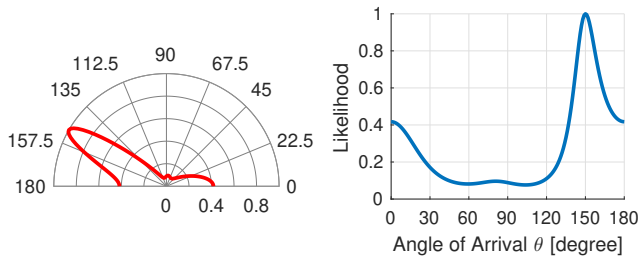
From Eq. (1), we get that the phase difference between any two adjacent antennas, denoted by ϕ, caused by a signal coming from AoA θ is equal to

$$\phi = \Phi(\theta) = \angle H_m - \angle H_{m-1} = -2\pi d \cos\theta/\lambda,\ m = 2, \cdots, M, \tag{3}$$

where $\Phi(\theta)$ is defined as the function that calculates the phase difference observed by an antenna array for a signal coming from angle θ. Here we assume that $\angle H_m$ is the phase of the LOS channel toward antenna m. We will explain how to extract this information in §3.1. Also, for simplicity, we omit the sub-index of subcarrier k hereafter. By measuring the phase difference ϕ from the CSIs of different antennas, we can derive the AoA of a path as

$$\theta = \Theta(\phi) = \arccos\left(\frac{\angle H_m - \angle H_{m-1}}{-2\pi d/\lambda}\right), \tag{4}$$

where $\Theta(\phi)$ is similarly defined as the function that transforms the phase difference between two adjacent antennas ϕ to the incoming angle θ of a signal. The transformation function is illustrated in



(a) AoA spectrum as user at 157.5° (b) Angle likelihood of (a)

Figure 3: Angle error due to phase wrapping. (a) An example AoA spectrum of user located at 157.5°. (b) The likelihood around 0° is also high due to phase wrapping.

Fig. 2(a). If an AP can receive the clean signal without any noise or phase errors, we can correctly learn the AoA θ when the number of antennas is sufficiently high for analyzing the AoA spectrum and eliminating the multipath effect. However, in practice, the received signal is never perfectly clean. In reality, the CSI is typically disturbed by noise, synchronization errors or hardware imperfection. Assume that the phase difference measured by an AP is $\tilde{\phi} = \phi + \epsilon$, where ϕ is the theoretical phase difference and ϵ is a small error in the measurement. Then, this small error ϵ would introduce an error to AoA as follows:

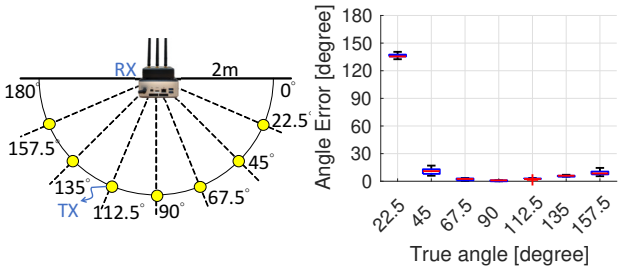
$$\tilde{\theta} = \Theta(\tilde{\phi}) = \arccos\left(\frac{\phi + \epsilon}{-2\pi d/\lambda}\right). \tag{5}$$

As $\arccos(\cdot)$ is not a linear function, a constant error in the phase difference could introduce unequal angle errors to $\tilde{\theta}$ for various locations, as shown in Fig. 2(a). In particular, the AoA error $\Delta\theta$ (in radians) caused by a small phase error ϵ can be found by taking the following derivative of the transformation function $\Theta(\phi)$:

$$\Delta\theta = \frac{d}{d\phi}\Theta(\phi) = \frac{d}{d\phi}\arccos\left(\frac{\phi}{-2\pi d/\lambda}\right) = \frac{\lambda}{2\pi d\sqrt{1 - \frac{\lambda^2\phi^2}{4\pi^2 d^2}}}, \tag{6}$$

which is illustrated in Fig. 2(b) (in degrees). We can observe from the figure that a constant phase error leads to a larger angle error when the true AoA is closer to either 0° and 180°. This phenomenon is conceptually the same with the theory of beamwidth variation subject to the same angular resolution in a uniform antenna array [24], i.e., a smaller 3dB beamwidth for central angles, while a larger beamwidth for side angles. Due to such unequal errors, the performance of AoA estimation is inherently more unreliable when a user is located around the boundary of an AP (i.e., 0° or 180°).

Even worse, since the phase difference is a sinusoidal function, if any two adjacent antennas are separated by the half wavelength ($d = \lambda/2$), the phase difference between any two adjacent antennas ranges from $-\pi$ (at 0°) to π (at 180°). As the phase difference in radians is typically wrapped to the interval of $[-\pi, \pi]$, the phase differences π and $-\pi$ are actually equivalent after wrapping. For example, when a client at a location corresponds to a phase close to π, its phase may be interfered by a small error ϵ such that the distorted phase $\pi + \epsilon$ will be wrapped to $-\pi + \epsilon$. That is to say, such phase wrapping ambiguity makes it difficult to distinguish between the AoAs of 0° and 180° and could lead to a huge AoA error for the boundary area even due to just a small phase error.



(a) Testbed (b) AoA Error distribution

true AoA	22.5	45	67.5	90	112.5	135	157.5
mean	137.5	9.2	2.0	0.7	2.8	6.0	9.3
std	9.5	1.7	0.7	0.3	0.2	0.3	2.2

(c) Mean and standard deviation of AoA errors

Figure 4: Empirical results of unequal AoA errors. (a) The testbed. (b) The error distribution of 150 packets in each location, which confirms high uncertainty in the boundary areas. (c) The boundary areas usually experience a higher AoA estimation error and a larger standard deviation, implying higher uncertainty.

Note that, if antennas are spaced differently, this wrapping ambiguity problem still exists but just in different AoAs. Fig. 3(a) demonstrates the AoA spectrum of a user locating at around 157.5° of an AP with the antenna space of $\lambda/2$. The results show that the amplitude of the angles around 0° is also large. If the AoA spectrum is directly used as the angle likelihood, as illustrated in Fig. 3(b), to locate a user, the localization error for users around the boundary area would be large due to such phase wrapping.

To verify the above observations, we conduct an experiment using commodity WiFi cards to measure the AoA errors for different known user locations toward a fixed AP, as illustrated in Fig. 4(a). The experiments are performed in a clean environment such that we can know the ground truth of the LOS AoA of any tested location. The AP collects the CSI of 150 packets in each location and adopts MUSIC to identify the peak AoA over time. Fig. 4(b) plots the AoA errors of the 150 packets, and Fig. 4(c) summarizes the mean and standard deviation of AoA errors. The results confirm that the areas closer to the boundary of an AP usually experience larger AoA errors. Also, the standard deviation of the boundary area is much larger than the center area, implying a lower confidence level of AoA measurements. As the number of APs grows, the likelihood of having an AP covering a user in its boundary area also increases. Any unreliable measure may offset the benefit of deploying more APs.

2.3 Variation of AoA Estimation

Another root cause of AoA errors is CSI variation. AoA estimation could be highly accurate when a transmitting client is static, e.g., laptops on the desk or steadily held smartphones. However, when a client is moving or even just slightly rotating the transmitting device, the CSIs would vary over time. We empirically compare the variation of AoA measurements when the client is static (on the desk) and dynamic (held with slight movement). Slight movement here refers to as a scenario where a user does not move (e.g., walking, running or driving) but just slightly shakes or moves

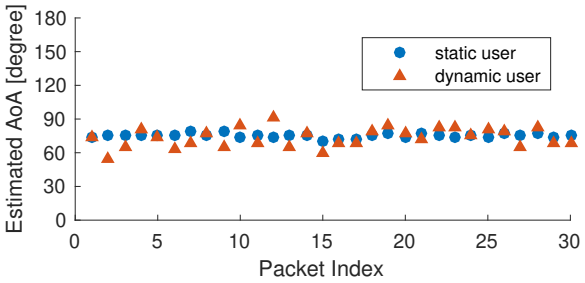


Figure 5: AoA dynamics. AoA estimates learned by MUSIC vary significantly in a dynamic environment. The angle errors can be up to 20.5°.

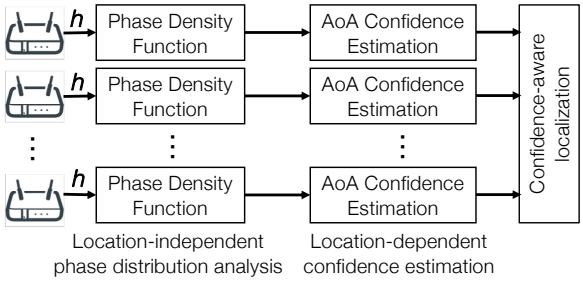


Figure 6: Framework of UAT. We decompose confidence inference to location-independent phase distribution analysis and location-dependent confidence transformation. The confidence levels of multiple APs are then used to enable reliable localization.

its device. In both scenarios, the client locates at 75° of the AP. The results shown in Fig. 5 demonstrate that, when the device is static, the AoAs estimated from different packets are quite stable. However, when the device has even a small movement, the AoA estimates can fluctuate significantly, as a result making it hard to use any single packet to locate the client. To achieve reliable localization, we should also consider this variation when inferring AoA confidence.

3 UAT Design

We now elaborate how UAT achieves confidence-aware localization. In UAT, multiple APs overhear packets sent by a target client. The APs then exploit the CSI tool [1][10] to extract the CSIs from the packets for MUSIC AoA estimation. As AoA uncertainty is determined by two factors, i.e., user location and channel dynamics, to better capture their impacts, we decouple confidence analysis to consider the two factors separately. That is, we analyze how the confidence level is affected by channel dynamics and user locations separately and integrate their effect for final localization, as illustrated in Fig. 6. To be more specific, as phase errors are irrelevant to user locations, UAT first analyzes the confidence levels of phase differences to characterize the effect of channel dynamics (§3.1). Next, UAT transforms the phase-domain confidence values to confidence of angle estimation with consideration of unequal angle biases (§3.2). Finally, the system leverages the confidence levels of APs as a hint to localize a target (§3.3).

3.1 Phase Density Function

To learn AoA reliability, we try to learn the probability density function of AoAs using sequential packets. As our main goal is to learn the confidence level of AoA measures, we can directly exploit MUSIC to first learn the AoA measures of sequential packets. MUSIC is a standard algorithm that decouples signals from multiple paths. Hence, we can obtain the initial LOS AoA measures without worrying about the multipath problem[1]. We then try to train the probability density function of those LOS AoA measures. In theory, a narrower density function implies a higher confidence level. However, as observed in Fig. 2 and Eq. (6), the errors in the angle domain (i.e., $\Delta\theta$) are not linear and may be large in boundary regions. That is, learning the distribution in the angle domain may be noisy. Fortunately, the errors in the phase domain (i.e., ϕ) are independent of user locations. With this observation, we propose to learn the density function of phase differences, rather than AoAs itself.

Let N denote the number of packets received by an AP, each of which outputs the CSIs of M antennas and K subcarriers, i.e., $\mathbf{H}_n = \{H_{n,m,k} : m = 1, \cdots, M, k = 1, \cdots, K, n = 1, \cdots, N\}$. Note that our goal is to learn the confidence of the LOS AoA. Hence, we should learn the phase difference introduced by the LOS path, instead of the phase difference of the raw CSIs, since the CSI is a superposition of the channels from multiple paths. To do so, for each packet n, we first perform AoA spectrum analysis and identify the peak as the LOS AoA, denoted by θ_n. The traditional MUSIC scheme is built on a narrowband channel and needs multiple packets to analyze the AoA spectrum. However, as we can now extract CSI measures from multiple subcarriers in a WiFi channel, we adopt the solution similar to [14], which leverages the CSIs of multiple subcarriers of a wideband channel, to perform AoA spectrum analysis and extract the LOS AoA. On the other hand, several previous efforts, e.g., [43] and [14], have investigated how to leverage a large number of antennas or multiple subcarriers of a wideband channel to suppress the multipath effect. Those designs can be incorporated with our system to refine the accuracy of initial LOS AoA extraction.

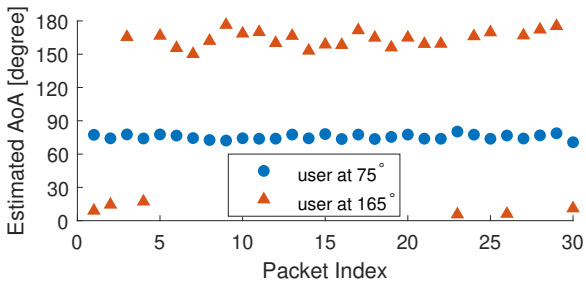
We then collect a set of potential LOS AoAs as $\mathcal{A} = \{\theta_n : 1 \le n \le N\}$. To eliminate the location-dependent bias, we transform the set of potential LOS AoAs to a set of phase differences:

$$\mathcal{P} = \{\phi_n = -2\pi d \cos(\theta_n)/\lambda : 1 \le n \le N\}. \quad (7)$$

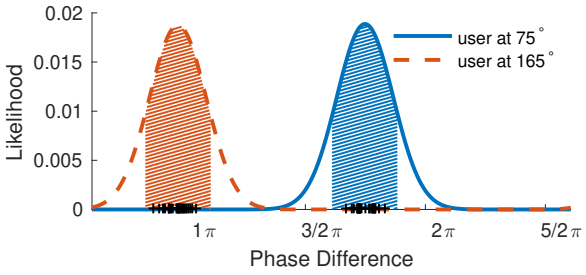
By such transformation, we get the phase difference of the LOS path, instead of that of the raw CSIs.

To learn the reliability of multiple packets, we aim at training the distribution of the measured LOS phase differences. Intuitively, if the distribution is more concentrated, it means that most packets output similar phase measurements, which should be highly confident estimations. However, if the measurement from different packets are spread, we should lower the confidence level for those estimates. We adopt a statistical technique, called *kernel density estimation* (KDE), to learn the distribution of phase differences. The main reason we pick KDE is that phase values are cyclic, ranging between $-\pi$ and π. For some client locations, the phase differences

[1]There exist many advanced AoA estimation algorithms proposed recently. We can alternatively exploit them (instead of MUSIC) to estimate initial LOS AoAs.



(a) AoA distribution (learned by MUSIC)



(b) Phase difference distribution

Figure 7: Phase-domain KDE of different locations. (a) A static client can have very different AoA variation in different locations. (b) The phase-domain KDE function of two locations however is very similar. The confidence interval (plotted as the colored region) of two locations is hence mostly the same. (The markers '+' on the x-axis are the phase differences corresponding to the AoA samples plotted in (a).)

may fluctuate around $-\pi$ and π due to phase wrapping. In those cases, the measured samples could be distributed in two intervals, one close to $-\pi$ and the other close to π (see Fig. 11 in §4). KDE is a non-parametric approach that estimates the probability density function of a random variable distributed in one or multiple intervals. It leverages an idea similar to *histograms*, but can characterize the properties of smoothness and continuity using a suitable *kernel function* such as triangular, biweight or normal. With this nice property, we can more accurately learn the distribution of phase differences of different intervals.

Say the phase differences $\phi_1, \phi_2, \cdots, \phi_N$ are i.i.d. univariate random samples from some distribution with an unknown density function $f_{\mathcal{P}}$. Then, the phase-domain kernel density estimator of $f_{\mathcal{P}}$ can be expressed by

$$\hat{f}_{\mathcal{P}}(\phi) = \frac{1}{NB} \sum_{n=1}^{N} \mathcal{K}\left(\frac{\phi - \phi_n}{B}\right), \quad (8)$$

where $\mathcal{K}$ is the kernel function, which is a non-negative function that integrates to one, and B is a smoothing parameter called the *bandwidth*. In our implementation, we use the normal distribution as the kernel function $\mathcal{K}(\cdot)$ in the phase domain. The trained KDE function estimates the likelihood of a phase difference, in which the peak ϕ^* can be deemed as the most likely phase difference estimated from multiple packets. The higher likelihood of the peak usually comes with less dispersion of the KDE function, meaning that we have higher confidence in this peak as most of phase difference samples are close to the peak.

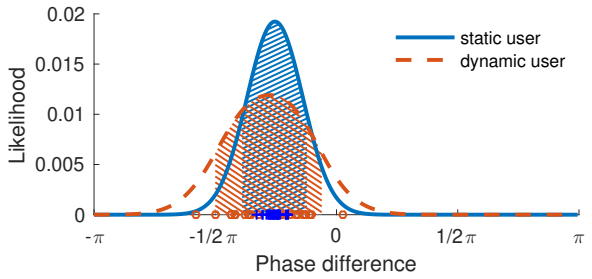


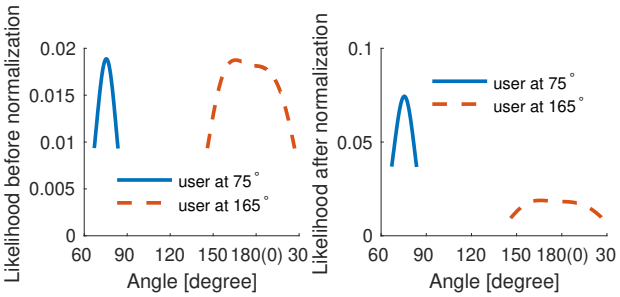
Figure 8: Phase-domain KDE for static and dynamic scenarios. Even for the same location, the phase-domain KDE functions for static and dynamic clients differ a lot. The confidence level of the dynamic case is much lower than that of the static case. (The markers '+' and 'o' on the x-axis are the samples of phase differences for the static and dynamic case, respectively.)

Fig. 7(a) illustrates the peak AoAs learned from sequential packets when a user locates at 75° and 165°, respectively. The figure shows that the AoAs distribute more divergently at 165° since the boundary region typically leads to a larger angle error. However, when those AoAs are converted back to the phase domain, the phase difference distributions for the two locations, as illustrated in Fig. 7(b), are actually quite similar as phase errors are only relevant to CSI variation, which comes from channel dynamics and hardware imperfection, but is independent of client locations. To verify this argument, we further convert the AoA distributions of a static client and a mobile client (i.e., Fig. 5) to their phase difference distributions, as illustrated in Fig. 8. The results demonstrate that, even though the client fixes their location in the two cases, the KDE function indeed becomes dispersed when the CSI variation increases due to movement. The above motivating examples show that, as the phase errors are independent of user locations, statistical dispersion of a phase-domain KDE function can be used as an indicator to infer confidence of multiple AoA measurements, regardless of client locations.

To keep only those samples with high enough confidence (i.e., likelihood), we leverage thresholding to filter out those mostly unlikely values in the phase-domain KDE function. For example, we could use half of the peak likelihood as the threshold to find a confidence interval, as the blue/red regions illustrated in Fig. 7(b) and Fig. 8. Only those phase differences within the confidence interval will be considered in the positioning algorithm. With such filtering, a more dispersed function generates a wider confidence interval, indicating lower confidence.

3.2 AoA Confidence Estimation

Given the phase-domain density function, we now know the likelihood of each phase difference, which is mainly determined by environmental dynamics, instead of user locations. However, as discussed in §2, since, theoretically, an AP has different confidence levels of AoA estimation for various areas, a client located in the boundary region of an AP may get an unreliable AoA estimation. To learn the confidence of an AP's AoA estimation, we should now transform the phase-domain kernel density function $\hat{f}_{\mathcal{P}}$ back to the angle-domain density function $\hat{f}_{\mathcal{A}}$. Since there exists an one-to-one mapping between a phase difference ϕ and an AoA $\theta = \Theta(\phi)$, as derived in Eq. (4), we can directly define the likelihood of a phase



(a) Converted AoA PDF (b) AoA PDF after normalization

Figure 9: Angle-domain density function. (a) The density function of the boundary area (165°) before normalization becomes wider due to a larger phase-to-angle error. (b) After normalization, the peak of the density function for the boundary area drops due to a wider confidence interval, verifying low confidence of AoA estimation.

difference ϕ as the likelihood of its corresponding AoA θ, i.e.,

$$f_{\mathcal{A}}(\theta) = \begin{cases} \hat{f}_{\mathcal{P}}(\Phi(\theta)) = \hat{f}_{\mathcal{P}}(-2\pi d \cos(\theta)/\lambda) & \phi_{\min} \le \Phi(\theta) \le \phi_{\max} \\ 0 & \text{otherwise} \end{cases}, \quad (9)$$

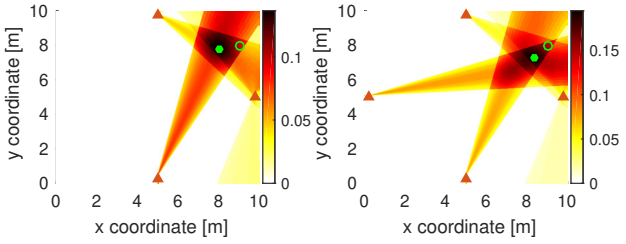
where $\phi_{\min}$ and $\phi_{\max}$, respectively, are the lower bound and upper bound of the confidence interval identified from the phase-domain KDE function. However, after transformation, $f_{\mathcal{A}}(\theta)$ is still not yet a probability density function since the integral over the entire space is not equal to one. We hence further normalize $f_{\mathcal{A}}(\theta)$ to get the angle-domain probability density function $\hat{f}_{\mathcal{A}}(\theta)$.

Fig. 9(a) illustrates the angle-domain functions transformed from the phase-domain density functions shown in Fig. 7, and Fig. 9(b) plots the angle-domain probability density functions after normalization. The figures show that, while the two locations (at 75° and 165°) have similar phase-domain density functions as phase errors are independent of locations, they do result in very different angle-domain density functions since a constant phase error introduces different angle errors for various client locations. As a result, the confidence interval after transformation could be enlarged or shrunk. As shown in Fig. 9, the confidence interval for a boundary area would become wider due to a larger phase-to-angle error. Therefore, due to lower confidence and higher variation, the peak likelihood of AoA in a boundary area (e.g., 165°) after normalization will also be reduced.

To sum up, though two locations may have the same phase-domain density function if a client sends packets in a similar condition (static or mobile), they produce different AoA distributions after phase-to-angle transformation. Hence, by combining KDE training and phase-to-angle transformation, we can consider channel variation and unequal angle errors separately and, hence, characterize AoA confidence more accurately.

3.3 Confidence-Aware Localization

Since the APs may have heterogeneous confidence, we propose a confidence-aware sector-based localization algorithm that combines AoA estimation of multiple APs to localize a client. Let S be the set of APs and $\omega_s = [\theta_{s,\min}, \theta_{s,\max}]$ denote the AoA-domain confidence interval of AP $s \in S$. Our angle-domain density function only assigns an angle within ω_s a likelihood. That is, AP s believes



(a) Three APs (b) Four APs

Figure 10: Heatmaps of likelihood estimation. The solid circle is the location estimated by UAT when the client locates in the hollow circle. The client is in the dead zone of the AP attached to the right wall, which corresponds to a wider sector (in which each direction is assigned a lower likelihood). The other APs have a narrow and confident sector.

that the client would locate in any location within the sector of size $|\omega_s|$ (i.e., confidence interval).

As mentioned above, an AP has a higher confidence level (smaller sector size $|\omega|$) if *i)* its received packets are stable and *ii)* its estimated peak AoA is not from a boundary area. A more confident AP has a narrower sector and assigns a higher likelihood to each potential location within the sector, while a less confident AP contributes a lower weight to each potential location since it assigns a likelihood to each direction of a larger sector, as the heatmaps shown in Fig. 10. By doing this, we naturally exploit the angle-domain density function to prioritize APs based on their confidence levels.

Given the likelihood estimation of APs, we can compute the likelihood of the client being at location $\mathbf{x}$, $L(\mathbf{x})$, by

$$L(\mathbf{x}) = \prod_{s=1,\cdots,S} \hat{f}_{\mathcal{A}}(\theta_s), \tag{10}$$

where θ_s is the angle from the candidate location $\mathbf{x}$ to AP s.[2] The target location can be the one with the maximal likelihood, i.e.,

$$\mathbf{x}^* = \arg\max L(\mathbf{x}), \tag{11}$$

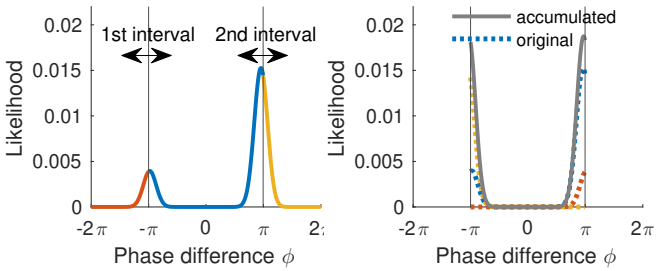
or, alternatively, the average coordinate of the locations with the top-K likelihoods. As each AP assigns the likelihood of each location based on its confidence, we enable multiple APs to improve reliability by unequal AoA sampling.

4 Practical Issues

This section discusses several practical issues.

CFO and SFO calibration. As clients and APs are not synchronized, the received signal phase would be affected by both the carrier frequency offset (CFO) and the sampling frequency offset (SFO). CFO, defined as $f_\Delta = f_{tx} - f_{rx}$, occurs because any two radios typically do not have exactly the same carrier frequency, i.e., $f_{tx} \neq f_{rx}$. Such a difference, even small, would accumulate quickly over time and add different offsets to different packets. On the other hand, SFO, denoted by ϵ, is caused by the fact that two radios may have slightly different sampling intervals T_{tx} and T_{rx}, respectively. To be specific, though, in theory, WiFi operates at 20MHz (or 40MHz), a transmitter and a receiver use different clocks and may

[2]In our angle-domain density function, the likelihoods of the locations outside a sector equal 0, which could make the likelihood multiplication become zero for many locations. To avoid this, we instead allocate a fairly small likelihood to the locations outside the sector.



(a) KDE before wrapping (b) KDE after wrapping

Figure 11: KDE wrapping. (a) Due to cyclic phase values, KDE may generate a function with two intervals. (b) We wrap the KDE function and get its accumulated function within the range of $[-\pi, \pi]$.

lead to unequal sampling intervals. The CSI with the additional phase offsets can be mathematically written as

$$H'_{m,k}(t) = H_{m,k}(t) * e^{-2j\pi(f_k + f_\Delta)(t+\epsilon)}, \tag{12}$$

where $H'_{m,k}$ is the actual CSI measured by antenna m over subcarrier k and $H_{m,k}$ is the ideal CSI over the air. Note that, for a multi-antenna AP, all the antennas are connected to the same clock and the same down-converter frequency and, hence, see the same SFO and CFO. Hence, we instead remove the offsets by simply taking the phase difference of the CSIs received by any two antennas as follows:

$$[1 \ \frac{H'_{2,k}}{H'_{1,k}} \ \frac{H'_{3,k}}{H'_{1,k}} \ \cdots \ \frac{H'_{M,k}}{H'_{1,k}}], \tag{13}$$

for each subcarrier $k = 1, \cdots, K$. After such division, the phase offsets can be canceled out, and $\angle\frac{H'_{m,k}}{H'_{1,k}}, m = 2, \cdots, M$ keeps only the phase difference caused by link propagation delay required for AoA estimation.

Initial phase calibration. While the antennas at an AP are tightly synchronized, they however may start at random initial phases, denoted by $\varphi_{m,k}$ for antenna m over subcarrier k, generated by the phased-locked loop (PLL). Hence, even after CFO and SFO calibration, there may still be a residual random initial phase as follows:

$$\angle\frac{H'_{m,k}}{H'_{1,k}} = \angle\frac{H_{m,k}}{H_{1,k}} + \Delta\varphi_{m,k}, m = 2, \cdots, M, k = 1, \cdots, K, \tag{14}$$

where $\Delta\varphi_{m,k} = (\varphi_{m,k} - \varphi^{tx}_k) - (\varphi_{1,k} - \varphi^{tx}_k) = \varphi_{m,k} - \varphi_{1,k}$. Note that this initial random phase difference is a constant no matter which client is transmitting since the initial phase of a client φ^{tx}_k is canceled out after taking CSI division. Also, this initial phase difference is independent of user location. With this property, we propose to leverage a reference location to remove such an offset. We notice that an AP should receive signals of the same phase at its multiple antennas if the signal comes from 90° (i.e., $\theta = \pi/2$). Hence, for a client locating at 90°, the phase difference remained in the measured CSI must be the initial phase differences, i.e.,

$$\angle\frac{H'^{\pi/2}_{m,k}}{H'^{\pi/2}_{1,k}} \equiv \Delta\varphi_{m,k}, m = 2, \cdots, M. \tag{15}$$

(a) AP (b) client

Figure 12: Prototype. The APs are equipped with Intel 5300 with three antennas. The client uses a commodity WiFi card equipped with an external antenna.

Thus, when an AP starts operating, we ask the operator to collect packets at a reference location along 90° for the AP to learn its initial phase differences $\Delta\varphi_{m,k}$. We finally use the following pre-processed CSIs as the inputs of MUSIC:

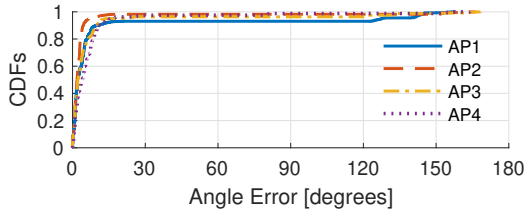
$$[1 \ \frac{H'_{2,k}}{H'_{1,k}} * e^{-j\Delta\varphi_{2,k}} \ \cdots \ \frac{H'_{M,k}}{H'_{1,k}} * e^{-j\Delta\varphi_{M,k}}], k = 1, \cdots, K. \quad (16)$$

Phase wrapping. Recall that phases are circular and usually wrapped to a value in between $-\pi$ and π. If a location creates a phase difference close to π (or $-\pi$), due to channel variation, the measured phase differences could distribute in two intervals, one close to π and the other close to $-\pi$, as the triangle markers illustrated in Fig. 7(a). Intuitively, we should wrap the samples in one interval to the other, e.g., $\phi = \phi + 2\pi$ for those ϕ close to $-\pi$. However, as the channel is noisy and highly variant, phase differences may be randomly distributed across $[-\pi, \pi]$, making it not easy to decide whether to wrap or not. Fortunately, KDE leverages a concept similar to *histograms* to learn the distribution. It hence can output the distribution of two or even more intervals if samples are distributed into multiple clusters. The curve shown in Fig. 11(a) illustrates the KDE function learned from the samples in Fig. 7(a) (triangle markers). This KDE function has two peaks close to $-\pi$ and π, respectively. Hence, instead of wrapping the phase differences of samples, we alternatively wrap the KDE likelihood function of phase differences, as the dotted lines shown in Fig. 11(b). After curve wrapping, we get the accumulated KDE function by transforming the original KDE function into

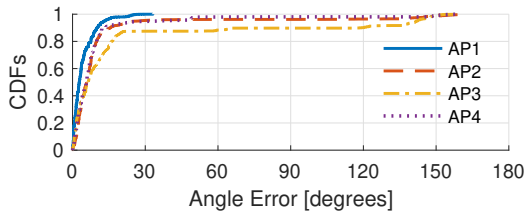
$$\hat{f}_{\mathcal{P}}(\phi) = \begin{cases} \hat{f}_{\mathcal{P}}(\phi) + \hat{f}_{\mathcal{P}}(\phi - 2\pi) + \hat{f}_{\mathcal{P}}(\phi + 2\pi) & \text{if } -\pi \le \phi \le \pi \\ 0 & \text{otherwise} \end{cases}.$$

5 Implementation

We deploy aBeing One mini PCs equipped with an off-the-shelf Intel 5300 WiFi NIC as the APs (receivers), as in Fig. 12(a), and use Lenovo X200s with Intel 5300 as a transmitting client, as in Fig. 12(b). All the nodes run Ubuntu Linux of version 14.04.4. The client and APs operate over channel 120 of the 5GHz WiFi band. To further extend the bandwidth to 40MHz, we set the HT40- parameter to bind channels 119 and 120. The client sends packets continuously with an inter-packet time of 25 *ms*. We install three antennas at each AP with spacing of the half wavelength and only equip one antenna at the client. The APs collect the traces of CSI



(a) Lobby



(b) Conference room

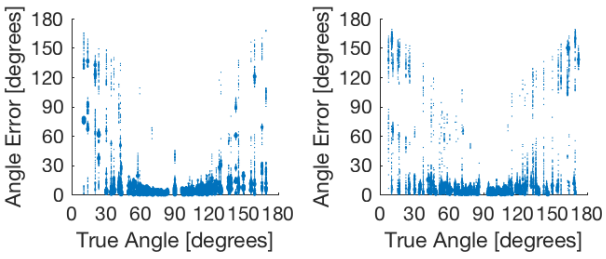
Figure 13: CDFs of AoA errors by MUSIC. While MUSIC produces accurate estimation for most of client locations, it however introduces an extremely large error for some critical locations.

using the 802.11 CSI Tool [10][1] and forward them to a back-end server, which executes the localization algorithm offline, including phase calibration, MUSIC and KDE, implemented in C++. In our experience, KDE training is not that sensitive to the settings of the bandwidth and the threshold. We hence fix the bandwidth and the threshold of KDE to 20 and 0.5, respectively, by default.

We run our experiments in two environments, a lobby and a conference room, which are typical rich multipath environments. The size of the lobby is $10m \times 10m$, while the size of the meeting room is about $10m \times 6.8m$. The lobby is a large open space with people walking around, while the meeting room is an enclosed space with objects like chairs and desks. In each environment, four UAT APs are deployed at the four boundaries of the field. The environments have ambient WiFi traffic. Unless otherwise stated, we use the CSIs of 20 packets to localize the target client. Empirically, collecting 20 packets is for balancing the tradeoff between the localization latency and accuracy. Our experiments are conducted in naturally dynamic environments with users walking through randomly. We evaluate the performance when the client is either static or mobile in different randomly-selected locations or trials, respectively. The reported results are the average or CDFs of multiple rounds of experiments.

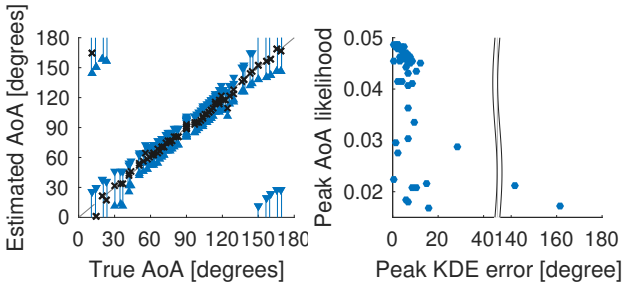
6 Results

We evaluate UAT in our testbed. As UAT is a system designed to quantify the reliability of AoA measures, it can be integrated with any AoA estimation algorithm, which estimates initial LOS AoA measures. In this evaluation, we pick ArrayTrack [43] as our initial AoA estimation algorithm. Hence, we compare our design with ArrayTrack [43] to check the effectiveness of considering reliability of AoA measures in a localization system. We further compare with a simple intersection scheme, which also uses ArrayTrack to estimate AoA measures but simply identifies the peak AoAs of multiple APs and finds the intersections of the directional beams along



(a) Lobby (b) Conference room

Figure 14: Distribution of MUSIC AoA errors. The AoA errors of the locations in the center area are usually small. However, the locations closer to 0 and 180 degrees get a larger AoA error, which can be as large as 168.6°.



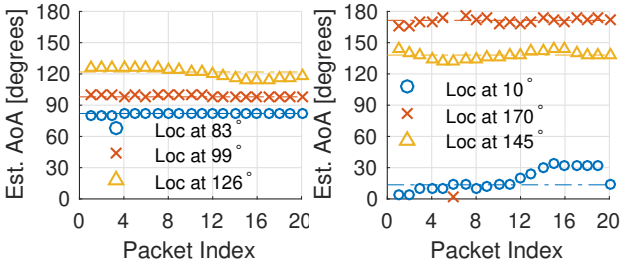
(a) AoA confidence interval (b) AoA errors vs. likelihood

Figure 15: AoA confidence via KDE training. (a) By KDE training and phase-to-angle transformation, we effectively enlarge the confidence interval of locations in the boundary areas. (b) UAT allocates a lower likelihood to AoA estimates with a larger error.

the peak AoAs of any two APs. It then outputs the average coordinate of those intersections as the target client location. ArrayTrack is an AoA-based localization system, which does not consider unequal angle reliability. It leverages a large number of antennas and sequential packets to suppress the multi-path effect. As multipath suppression is out of the scope of this work, we hence exclude the design of multipath suppression in all the comparison schemes. Similarly, for fair comparison, we use 3-antenna APs but leverage the CSIs of multiple subcarriers to enhance AoA spectrum analysis resolution for all the comparison schemes. The experiments are mainly designed to verify the effectiveness of unequal AoA tracking. There have been many recent solutions for improving localization resolution, which are not all implemented in our evaluation. We believe that the benefit of UAT's confidence-aware design can further enhance the tail performance of those advanced works.

6.1 Macro Benchmark

MUSIC AoA error. We first check the errors of the peak AoA directly estimated by MUSIC, instead of our KDE training. We deploy four APs in this experiment and place a dynamic client in randomly-selected locations. Fig. 13 illustrates the CDFs of the absolute AoA errors for various APs. We can observe from the figures that, though most of APs can achieve very accurate estimation for most of locations, they however could introduce an AoA error



(a) Center Area (b) Boundary Area

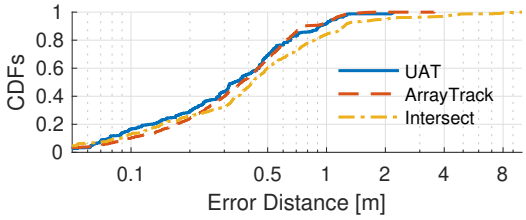
Figure 16: MUSIC angle errors across packets and the peak AoA in KDE. AoAs estimated by MUSIC could fluctuate across packets, especially for the boundary areas. UAT characterizes the phase difference distribution more accurately and, as compared to traditional MUSIC, achieves better accuracy.

larger than 30 degrees for some critical locations. Most of large errors are contributed by the locations belonging to the boundary area of some APs. If those uncertain estimates are used for localizing a client, the positioning error would be large.

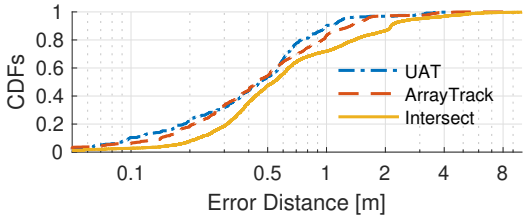
To dig into the distribution of AoA errors, we further illustrate in Fig. 14 the scatter plot of the MUSIC AoA errors for various true angles of client locations. In the scatter plot, we use the size of markers to represent the percentage of samples that produce this error. That is, a larger marker means an error happening more frequently, while a smaller marker denotes a less frequent error. The results confirm that most of uncertain estimates, though not frequent, are from those boundary areas, where a small phase error could lead to a significant angle error. The localization performance can be improved if the system can lower the confidence of an AP for a client located in its dead zones.

KDE AoA estimation. This experiment examines whether KDE can capture the unequal reliability of AoA estimation. We deploy four APs and move the client to randomly selected locations. Fig. 15(a) plots the confidence interval of AoA estimates and the peak AoA estimated by KDE (represented as markers '×') for various client locations. The results show that, as expected, by limiting the confidence interval of the phase difference distribution and transforming it to the angle domain, we obtain different sizes of the angle confidence intervals for various locations. The locations closer to the boundary area have a larger confidence interval, implying a lower likelihood and thereby a lower weight assigned to those estimates. By contrast, the confidence intervals of central locations are much smaller. Hence, the angle estimates of the packets from the center area have a higher likelihood, which reflects higher confidence for localization. On the other hand, we can observe that those confidence intervals can mostly cover the true angle of client locations, i.e., the dashed line with the slope of 1. It indicates that UAT adapts the size of the confidence interval suitably to filter out the measures far from the ground truth.

We next verify whether KDE training can capture AoA estimation confidence correctly. Intuitively, for reliable localization, an AoA estimate with a larger error should be allocated a smaller likelihood. To confirm this, in Fig. 15(b), we plot the relationship

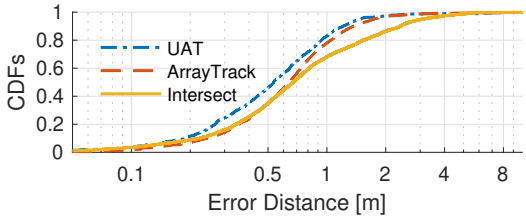


(a) Lobby

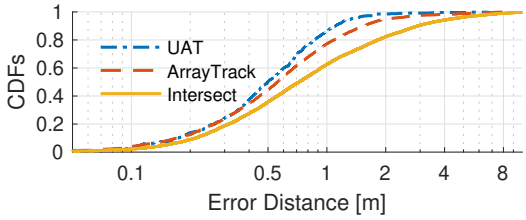


(b) Conference room

Figure 17: Localization errors in the static scenario. When the client is completely static, the CSIs of packets are fairly static. ArrayTrack hence performs comparable to our UAT, except for some critical locations belonging to the boundary area.



(a) Lobby



(b) Conference room

Figure 18: Localization errors in the dynamic scenario. When the client has subtle movement, the CSIs could fluctuate slightly. UAT's KDE infers AoA reliability and reduces localization errors.

between the estimation error of the peak AoA of the KDE function and its assigned likelihood. It can be observed that, for those AoA estimates with an error smaller than 20 degrees, UAT usually allocates those samples a high likelihood, which means that those samples can be effectively used to localize a client. However, for other AoA estimates corresponding to a larger error (e.g., larger than 30 degrees), UAT's KDE identifies their uncertainty and lowers their likelihood such that those unreliable measures would less likely bias the localization decision. This figure shows that KDE correctly characterizes AoA reliability, allowing us to enable confidence-aware localization.

We then check how well KDE performs in different areas. Fig. 16(a) plots the LOS AoAs of 20 packets estimated by conventional MUSIC in three locations of the center area, along 83, 99 and 126 degrees of an AP, respectively. The results show that the AoA estimates of the locations in the center area are fairly stable. We then plot the peak learned by KDE as the dashed horizontal lines, which can be very close to the ground truth AoA. Fig. 16(b), however, shows that the estimated AoAs of 20 packets in three locations of the boundary area, along 10, 145 and 170 degrees of an AP, respectively, are spread due to a higher phase-to-angle error. If we simply learn the angle distribution from those samples, we may get a peak far from the true angle. For example, for the location at 10 degrees in Fig. 16(b), the measures are distributed between 0 and 30 degrees. Simply learning the distribution of those AoAs may output a peak far from the ground truth. However, UAT's KDE accurately learns the distribution in the phase domain, which is location independent, and is able to find an accurate peak angle after phase-to-angle transformation.

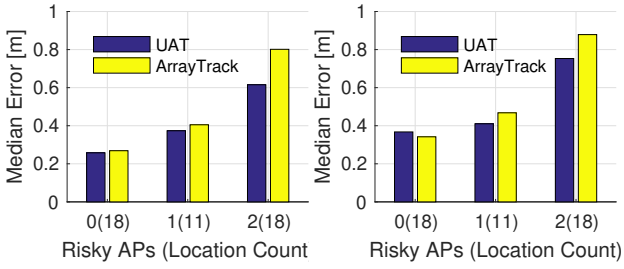
6.2 Performance Comparison

We next compare the localization performance of UAT with the comparison schemes in the static, dynamic and mobile scenarios, respectively. In this experiment, we will show the localization accuracy of all the possible locations (instead of only those in the boundary areas) in order to demonstrate the benefit of UAT especially for risky locations.

Static scenario. In this experiment, we configure four APs attached to the four walls in the testbed. A client is statically deployed in 50 randomly-selected locations. For each location, we let the client send 20 packets for the APs to learn the phase difference distribution based on KDE. The server then calculates the AoA density function of the APs and performs confidence-aware localization.

Figs. 17(a) and 17(b) plot the CDFs of the localization errors in the lobby and conference room, respectively. The intersection scheme only uses the peak AoA but does not consider estimation likelihood. It hence introduces a larger AoA error, which in turn worsens localization accuracy. The results also show that, when the client is static, the CSIs are relatively stable. Hence, even without leveraging confidence inference, ArrayTrack can achieve a performance similar to UAT in most of locations. The main reason is that, as a location is within the safe area of all the neighboring APs, the system can obtain a fairly good localization results even if it does not explicitly consider the confidence level of AoA measures. However, we can still observe from the figures that UAT achieves a smaller localization error than ArrayTrack in some challenging locations, where some APs may create a larger AoA estimation error. Specifically, in UAT, 90% of locations can achieve sub-meter level accuracy, while, in ArrayTrack, only 80% of locations get an error less than 1 m in the conference room. This verifies the benefit of UAT's reliability-aware design for tail performance enhancement.

Dynamic scenario. We also perform the same experiment, but ask the client to hold the device with subtle movement. The purpose of this experiment is to check how the variation of CSI caused by device movement affects the localization performance. We again compare the CDFs of the localization errors of different



(a) Static (b) Dynamic

Figure 19: Localization errors in different areas. UAT adapts the confidence levels of APs and is especially effective for the locations covered by more risky APs.

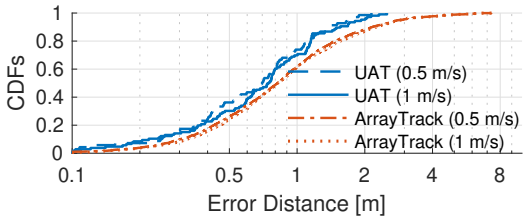
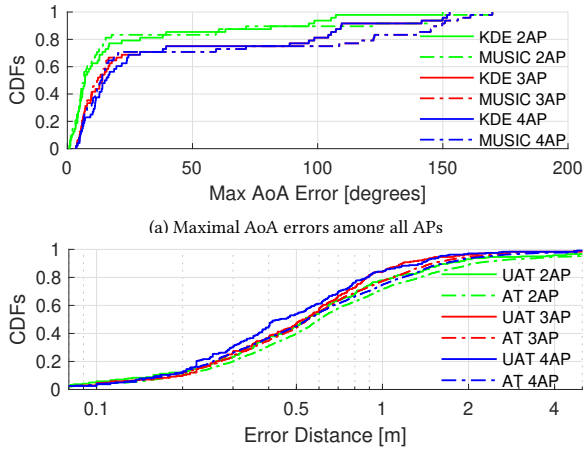


Figure 20: Localization errors in the mobile scenario. CSI variation becomes larger in the mobile scenario. By KDE training, UAT filters out unreliable CSIs and produces a localization error smaller than 2 m for most of locations.

schemes in the lobby and conference room, respectively, as shown in Figs. 18(a) and 18(b). The results show that, as now the CSI is more noisy, ArrayTrack does not prioritize AoA measures and hence produces a larger error. However, UAT learns the AoA density function based on KDE. It can hence filter out some unreliable measures and obtain an AoA density function that is more likely to cover the true AoA. Comparing to the results in the static scenario (Fig. 17), UAT benefits more from considering AoA variation and reduces the 75-percentile error from 0.96 m to 0.77 m. Also, UAT is especially effective for challenging locations. The 90-percentile localization error can be reduced by 27.5% and 62.1% as compared to ArrayTrack and Intersection, respectively.

Impact of unequal reliability. To take a closer look at where the improvement comes from, we classify the client locations into three groups: 0, 1 and 2 risky APs. A risky AP means that the client location is within the boundary area of this AP. As there exists no clear boundary of risky area, in our experiments, we define the range of $[0, 45]$ and $[135, 180]$ degrees as the boundary area of an AP. Hence, a client location has i risky APs if it locates within the boundary area of i APs. A location corresponding to more risky APs usually gets a larger localization error due to unequal AoA reliability.

We illustrate the median error of the three groups in Figs. 19(a) and 19(b) for the static and dynamic scenarios, respectively. The results show that, in general, a location experiences a higher localization error if it is covered by more risky APs due to unreliable AoA estimation, verifying our motivation. For the group with no risky AP (i.e., locating in the safe area of all the APs), UAT actually performs similar to ArrayTrack since they essentially both follow



(a) Maximal AoA errors among all APs

(b) Localization Errors

Figure 21: Impact of number of APs. A client is more likely to be covered by an unreliable AP when there are more APs. UAT identifies those unreliable measurements and effectively exploits multiple APs to enhance localization accuracy.

MUSIC. However, the more risky APs covering a location, the more enhancement UAT can achieve, showing the effectiveness of our confidence-aware localization. Though the risky area in our definition is wide, UAT reduces the median localization error by 23.2% and 14.3% for the static and dynamic scenarios, respectively, when a location belongs to the boundary area of two risky APs. The improvement is in fact more significant for users located closer to 0 and 180 degrees of a risky AP.

Mobile scenario. In this experiment, we ask the client to walk along a straight line toward a randomly selected direction. We try two walking speeds, around 0.5 m/s and 1 m/s, respectively. For each walking speed, we repeat the experiment 10 times. As it is harder to obtain the ground truth location during mobility, we tag the ground truth locations of the beginning and ending points of a trajectory and assume that the client walks at a constant speed. The ground truth location during walking is then estimated by interpolation.

Fig. 20 compares the CDFs of localization errors of UAT and ArrayTrack during walking. The results show that the errors become larger when the walking speed increases, since the CSI variation gets larger when the client has higher mobility. Hence, the phase estimates of different packets may differ a lot, leading to lower reliability. However, UAT leverages multiple packets to find a phase distribution fitting the ideal phase difference and, hence, is more resistant to random noise. As compared to ArrayTrack, we reduce the median localization error from 0.82 m to 0.75 m. For challenging locations, we can improve the accuracy significantly and reduce the 90-percentile localization error by 28.4%.

Impact of number of APs. We finally evaluate the impact of the number of APs. In this experiment, we ask a dynamic client to stand in 50 randomly-selected locations, and vary the number of APs from 2 to 4. We first show the CDFs of the maximal AoA error among all the APs for conventional MUSIC and our KDE in

Fig. 21(a). The figure shows that, when the number of APs grows, a client is more likely to be located in the boundary area of an AP, leading to a large AoA error. UAT's KDE scheme considers the distribution of multiple measurements and, thereby, reduces the maximal AoA error effectively. Fig. 21(b) then plots the CDFs of the localization errors of UAT and ArrayTrack, respectively. The results show that the CSIs of packets could fluctuate slightly and lower the confidence level of each AP. Installing the fourth AP, thus, helps further improve confidence of AoA estimation and decrease the final localization error. More importantly, as the probability of being covered by an unreliable AP increases as there are more APs, the gain of UAT over ArrayTrack gets higher when there are more APs deployed.

To summarize, we experimentally verify that AoA measures are vulnerable in the boundary areas and mobile scenarios. Our KDE-based estimation effectively learns the distribution of AoA measures and more reliably identifies the peak AoA for localization. With consideration of the AoA confidence level, UAT improves the tail location performance and allows 90% of locations to obtain sub-meter level accuracy.

7 Related Work

We classify related work into the following categories:

RSSI-based localization. Earlier studies [4–7, 11, 18, 19, 25, 28, 40, 44, 46, 48] mainly rely on the received signal strength indicator (RSSI) information to localize a target client. The RSSI-based approaches can further be divided into the fingerprinting-based method [5, 46] and the model-based method [4, 5, 28]. Some later work further fuses other techniques, e.g., acoustic signal [19, 22] or particle filtering [27], to improve positioning accuracy. The limitation of RSSI-based systems is that RSSI measures are highly variant, which makes the localization result highly uncertain. Recently, some systems, e.g., [32, 41, 45], further leverage CSI to minimize uncertainty and improve accuracy.

ToF-based localization. This type of localization systems calculates time-of-flight (ToF) of an LOS link and estimates the link distance accordingly [3, 9, 13, 16, 20, 21, 35, 36, 47]. As the bandwidth of WiFi is small, it is unlikely to obtain a sufficient resolution of ToF for WiFi-based localization. To address this limitation, WiTrack [3][2] instead uses low-power FMCW radar across 1.69 GHz of frequency bands to perform fine-grained ToF estimation. Chronos [35] then exploits a frequency-hopping technique to explore the entire WiFi frequency bands crossing from 2.4GHz to 5GHz. However, the frequency-hopping scheme requires clients to frequently switch among different channels, which may hinder normal data transmissions and could also introduce significant interfering signals to environments.

AoA-based localization. Some systems utilize antenna arrays to calculate the AoA of a target client [8, 12, 14, 15, 17, 23, 26, 29, 31, 33, 37, 39, 43]. With the development of MIMO technologies and the release of the CSI tool [1][10], AoA-based approaches become more and more popular because of its better accuracy. In ArrayTrack [43], 8-antenna antenna arrays are built using the WARP boards to implement the MUSIC algorithm. CUPID [31] utilizes commodity WiFi cards to estimate AoAs but jointly leverage distance estimation to refine localize accuracy. SpotFi [14] aims at combating the limited number of antennas in commodity WiFi chipsets by a super-resolution algorithm. Some research [15, 37] even uses synthetic aperture radar (SAR) to simulate an antenna array using only a small number of antennas. The aforementioned work does not explicitly consider unequal AoA reliability in localization. Hence, the localization results for the dead zones would be vulnerable. Though ArrayTrack mentions a lower reliability for the areas around 0° and 180°, it neither studies this phenomenon nor gives any solutions to it. The unequal reliability problem may be bypassed by using circular antenna arrays [34, 49]. However, circular arrays usually require special deployment and still suffer from unequal reliability in the z-axis. Our work demystifies the roots causing unequal AoA reliability and proposes a confidence-aware localization algorithm to tackle this problem and enhance the tail performance.

8 Conclusion and Future Directions

This paper introduces UAT, an AoA-based localization system that addresses the unequal reliability issue. We first conduct extensive measurement studies to characterize AoA estimation reliability and identify the root causes of unreliable AoA estimation. This work is the first that qualifies the reliability of AoA estimates with consideration of location-dependent phase-to-angle transformation errors. We then develop a confidence-aware localization system, UAT, which leverages phase-domain KDE and phase-to-angle transformation to enable sector-based localization. By confidence-aware sector size adaptation, UAT allows multiple APs to weigh their decisions automatically and produce highly-confident localization results. Our prototype implementation shows that considering unequal AoA reliability prevents risky APs from biasing the localization result. By UAT's adaptive sector-based localization, the median localization error of the risky area can be reduced by 23.2% and 14.3% for the static and dynamic scenarios, respectively, as compared to reliability-oblivious localization.

This work focuses on prioritizing AoA measurements based on their confidence levels for a given WiFi infrastructure. Some particular dead spots may still be poorly localized if they are within the risky areas of all the neighboring APs. To further avoid the dead spot problem, one can optimize the deployment of APs such that any location can be covered by a sufficient number of reliable APs. We leave investigating this issue for future research.

Acknowledgments

We thank Haitham Hassanieh and Fadel Adib for their insightful comments. We thank Peter Druschel for helping shepherd our paper, and thank anonymous reviewers for helping improve the presentation of this work. This research is partially supported by Delta Technology, MOST Pervasive Artificial Intelligence Research (PAIR) Labs under contract No. 105-2218-E-009-003 (MoST), 107-EC-17-A-22-I1-0010 (MoEA), 106-2628-E-009-004-MY3, 107-2221-E-009-023-MY2 and 107-2218-E-009 -048, and "Center for Open Intelligent Connectivity" (MoE).

References

[1] [n. d.]. Linux 802.11n CSI Tool. https://dhalperi.github.io/linux-80211n-csitool/.

[2] Fadel Adib, Zachary Kabelac, and Dina Katabi. 2015. Multi-person localization via RF body reflections. In *USENIX NSDI*.

[3] Fadel Adib, Zachary Kabelac, Dina Katabi, and Robert C Miller. 2014. 3D Tracking via Body Radio Reflections. In *USENIX NSDI*.

[4] Abdalkarim Awad, Thorsten Frunzke, and Falko Dressler. 2007. Adaptive distance estimation and localization in WSN using RSSI measures. In *IEEE Digital system design architectures, methods and tools (DSD)*.

[5] Paramvir Bahl and Venkata N Padmanabhan. 2000. RADAR: An in-building RF-based user location and tracking system. In *IEEE INFOCOM*.

[6] Paolo Barsocchi, Stefano Lenzi, Stefano Chessa, and Gaetano Giunta. 2009. A novel approach to indoor RSSI localization by automatic calibration of the wireless propagation model. In *IEEE VTC-Spring*.

[7] Long Cheng, Cheng-Dong Wu, and Yun-Zhou Zhang. 2011. Indoor robot localization based on wireless sensor networks. *IEEE Transactions on Consumer Electronics* 57, 3 (Aug. 2011).

[8] Jon Gjengset, Jie Xiong, Graeme McPhillips, and Kyle Jamieson. 2014. Phaser: Enabling phased array signal processing on commodity WiFi access points. In *ACM MobiCom*.

[9] Stuart A Golden and Steve S Bateman. 2007. Sensor measurements for Wi-Fi location with emphasis on time-of-arrival ranging. *IEEE Transactions on Mobile Computing* 6, 10 (Oct. 2007), 1185–1198.

[10] Daniel Halperin, Wenjun Hu, Anmol Sheth, and David Wetherall. 2011. Tool release: Gathering 802.11n traces with channel state information. *ACM SIGCOMM Computer Communication Review* 41, 1 (Jan. 2011), 53–53.

[11] Yan Huang, Jianying Zheng, Yang Xiao, and Miao Peng. 2015. Robust localization algorithm based on the RSSI ranging scope. *International Journal of Distributed Sensor Networks* 11, 2 (Feb. 2015), 587318.

[12] Daniele Inserra and Andrea M Tonello. 2013. A frequency-domain LOS angle-of-arrival estimation approach in multipath channels. *IEEE Transactions on Vehicular Technology* 62, 6 (July 2013), 2812–2818.

[13] Tufan C Karalar and Jan Rabaey. 2006. An RF ToF based ranging implementation for sensor networks. In *IEEE ICC*.

[14] Manikanta Kotaru, Kiran Joshi, Dinesh Bharadia, and Sachin Katti. 2015. Spotfi: Decimeter level localization using WiFi. In *ACM SIGCOMM*.

[15] Swarun Kumar, Stephanie Gil, Dina Katabi, and Daniela Rus. 2014. Accurate indoor localization with zero start-up cost. In *ACM MobiCom*.

[16] Steven Lanzisera, David Zats, and Kristofer SJ Pister. 2011. Radio frequency time-of-flight distance measurement for low-cost wireless sensor localization. *IEEE Sensors Journal* 11, 3 (March 2011), 837–845.

[17] Xiang Li, Shengjie Li, Daqing Zhang, Jie Xiong, Yasha Wang, and Hong Mei. 2016. Dynamic-MUSIC: accurate device-free indoor localization. In *ACM UbiComp*.

[18] H. Lim, L. . Kung, J. C. Hou, and H. Luo. 2006. Zero-Configuration, Robust Indoor Localization: Theory and Experimentation. In *IEEE INFOCOM*.

[19] Hongbo Liu, Yu Gan, Jie Yang, Simon Sidhom, Yan Wang, Yingying Chen, and Fan Ye. 2012. Push the limit of WiFi based localization for smartphones. In *ACM MobiCom*.

[20] Andreas Marcaletti, Maurizio Rea, Domenico Giustiniano, Vincent Lenders, and Aymen Fakhreddine. 2014. Filtering Noisy 802.11 Time-of-Flight Ranging Measurements. In *ACM CoNEXT*.

[21] Alex T Mariakakis, Souvik Sen, Jeongkeun Lee, and Kyu-Han Kim. 2014. Sail: Single access point-based indoor localization. In *ACM MobiSys*.

[22] Rajalakshmi Nandakumar, Krishna Kant Chintalapudi, and Venkata N Padmanabhan. 2012. Centaur: locating devices in an office environment. In *ACM MobiCom*.

[23] Dragoş Niculescu and Badri Nath. 2004. VOR Base Stations for Indoor 802.11 Positioning. In *ACM MobiCom*.

[24] Sophocles J Orfanidis. 2002. Electromagnetic waves and antennas. (2002).

[25] Anindya S Paul and Eric A Wan. 2009. RSSI-based indoor localization and tracking using sigma-point Kalman smoothers. *IEEE Journal of Selected Topics in Signal Processing* 3, 5 (Oct. 2009), 860–873.

[26] Kun Qian, Chenshu Wu, Zheng Yang, Zimu Zhou, Xu Wang, and Yunhao Liu. 2018. Enabling Phased Array Signal Processing for Mobile WiFi Devices. *IEEE Transactions on Mobile Computing* 17, 8 (Aug. 2018), 1820–1833.

[27] Anshul Rai, Krishna Kant Chintalapudi, Venkata N Padmanabhan, and Rijurekha Sen. 2012. Zee: Zero-effort crowdsourcing for indoor localization. In *ACM MobiCom*.

[28] Andreas Savvides, Chih-Chieh Han, and Mani B Strivastava. 2001. Dynamic fine-grained localization in ad-hoc networks of sensors. In *ACM MobiCom*.

[29] A. H. Sayed, A. Tarighat, and N. Khajehnouri. 2005. Network-based wireless location: challenges faced in developing techniques for accurate wireless location information. *IEEE Signal Processing Magazine* 22, 4 (July 2005), 24–40.

[30] Ralph Schmidt. 1986. Multiple emitter location and signal parameter estimation. *IEEE Transactions on Antennas and Propagation* 34, 3 (March 1986), 276–280.

[31] Souvik Sen, Jeongkeun Lee, Kyu-Han Kim, and Paul Congdon. 2013. Avoiding multipath to revive inbuilding WiFi localization. In *ACM MobiSys*.

[32] Souvik Sen, Božidar Radunovic, Romit Roy Choudhury, and Tom Minka. 2012. You are facing the Mona Lisa: spot localization using PHY layer information. In *ACM MobiSys*.

[33] Elahe Soltanaghaei, Avinash Kalyanaraman, and Kamin Whitehouse. 2018. Multipath Triangulation: Decimeter-level WiFi Localization and Orientation with a Single Unaided Receiver. In *ACM MobiSys*.

[34] H. Sun and Y. Lu. 2012. DOA estimation with a vector circular array. In *IEEE Asia-Pacific Conference on Antennas and Propagation*.

[35] Deepak Vasisht, Swarun Kumar, and Dina Katabi. 2016. Decimeter-Level Localization with a Single WiFi Access Point. In *USENIX NSDI*.

[36] Jie Wang, Qinghua Gao, Yan Yu, Xiao Zhang, and Xueyan Feng. 2016. Time and energy efficient TOF-based device-free wireless localization. *IEEE Transactions on Industrial Informatics* 12, 1 (Feb. 2016), 158–168.

[37] Jue Wang and Dina Katabi. 2013. Dude, where's my card?: RFID positioning that works with multipath and non-line of sight. In *ACM SIGCOMM*.

[38] X. Wang, L. Gao, and S. Mao. 2017. BiLoc: Bi-Modal Deep Learning for Indoor Localization With Commodity 5GHz WiFi. *IEEE Access* 5 (2017), 4209–4220.

[39] Yue Wang and KC Ho. 2015. An asymptotically efficient estimator in closed-form for 3-D AOA localization using a sensor network. *IEEE Transactions on Wireless Communications* 14, 12 (Dec. 2015), 6524–6535.

[40] Yixin Wang, Qiang Ye, Jie Cheng, and Lei Wang. 2015. RSSI-based bluetooth indoor localization. In *IEEE Mobile Ad-hoc and Sensor Networks (MSN)*.

[41] Kaishun Wu, Jiang Xiao, Youwen Yi, Min Gao, and Lionel M Ni. 2012. Fila: Fine-grained indoor localization. In *IEEE INFOCOM*.

[42] Yaxiong Xie, Jie Xiong, Mo Li, and Kyle Jamieson. 2018. mD-Track: Leveraging Multi-Dimensionality in Passive Indoor Wi-Fi Tracking. *ArXiv e-prints* arXiv:1812.03103 (2018).

[43] Jie Xiong and Kyle Jamieson. 2013. Arraytrack: a fine-grained indoor location system. In *USENIX NSDI*.

[44] Chenren Xu, Bernhard Firner, Robert S Moore, Yanyong Zhang, Wade Trappe, Richard Howard, Feixiong Zhang, and Ning An. 2013. SCPL: Indoor device-free multi-subject counting and localization using radio signal strength. In *IEEE IPSN*.

[45] Zheng Yang, Zimu Zhou, and Yunhao Liu. 2013. From RSSI to CSI: Indoor localization via channel response. *ACM Computing Surveys (CSUR)* 46, 2 (Dec. 2013), 25:1–25:32.

[46] Moustafa Youssef and Ashok Agrawala. 2005. The Horus WLAN location determination system. In *ACM MobiSys*.

[47] Moustafa Youssef, Adel Youssef, Chuck Rieger, Udaya Shankar, and Ashok Agrawala. 2006. Pinpoint: An asynchronous time-based location determination system. In *ACM MobiSys*.

[48] Giovanni Zanca, Francesco Zorzi, Andrea Zanella, and Michele Zorzi. 2008. Experimental comparison of RSSI-based localization algorithms for indoor wireless sensor networks. In *Proceedings of ACM workshop on Real-world wireless sensor networks*.

[49] Le Zuo and Jin Pan. 2017. Accurate 2-D AOA Estimation and Ambiguity Resolution for a Single Source under Fixed Uniform Circular Arrays. *International Journal of Antennas and Propagation* 2017 (12 2017), 1–6.