SpotFi: Decimeter Level Localization Using WiFi

Manikanta Kotaru, Kiran Joshi, Dinesh Bharadia, Sachin Katti
Stanford University
Stanford, CA, USA
{mkotaru, krjoshi, dineshb, skatti}@stanford.edu

ABSTRACT

This paper presents the design and implementation of SpotFi, an accurate indoor localization system that can be deployed on commodity WiFi infrastructure. SpotFi only uses information that is already exposed by WiFi chips and does not require any hardware or firmware changes, yet achieves the same accuracy as state-of-the-art localization systems. SpotFi makes two key technical contributions. First, SpotFi incorporates super-resolution algorithms that can accurately compute the angle of arrival (AoA) of multipath components even when the access point (AP) has only three antennas. Second, it incorporates novel filtering and estimation techniques to identify AoA of direct path between the localization target and AP by assigning values for each path depending on how likely the particular path is the direct path. Our experiments in a multipath rich indoor environment show that SpotFi achieves a median accuracy of 40 cm and is robust to indoor hindrances such as obstacles and multipath.

CCS Concepts

•**Information systems** → **Location based services;** *Sensor networks; Global positioning systems;* •**Networks** → **Location based services;**

Keywords

Indoor localization; Wireless; WiFi; OFDM; CSI

1. INTRODUCTION

Indoor localization systems using WiFi infrastructure should ideally satisfy the following three requirements:

- **Deployable**: They should be easily deployable on existing commodity WiFi infrastructure without requiring any hardware or firmware changes at the access points (APs); they should only work with information like RSSI and CSI (Channel State Information) that is already exposed by commodity, deployed APs.

- **Universal**: They should be able to localize any target device that has a commodity WiFi chip and nothing else. They should not require the target to have any other hardware, be it sensors such as accelerometers, gyroscopes, barometers, cameras, etc., or radios such as UWB, ultrasound, Bluetooth LE, etc.
- **Accurate**: They should be accurate, ideally as accurate as the best known localization systems that use wireless signals (even including those that do not satisfy the above two requirements). To the best of our knowledge, the most accurate such localization systems are ArrayTrack [1] and Ubicarse [2] and these systems achieve an accuracy ranging from 30–50 cm in office environments. Achieving such accuracy would be the target.

If the above three requirements are satisfied, we can imagine indoor localization becoming a ubiquitous service like GPS that can be installed on *already deployed WiFi infrastructure* and made available to any device with a WiFi chip.

However, to the best of our knowledge, no system that satisfies all three requirements exists. RSSI based systems are deployable and universal, but are not accurate; their median accuracy ranges from 2–4 m [3, 4, 5]. Recent techniques that rely on angle of arrival (AoA) estimation such as ArrayTrack and LTEye [6] are accurate and universal but are not deployable as they require hardware modifications. For example, ArrayTrack uses six to eight antennas whereas typical APs have three, and LTEye requires motorized rotating antennas which are not found on deployed APs. Other techniques that combine inputs from multiple sensors such as Ubicarse [2] are accurate and deployable but not universal; they require that the target device has access to other sensing modes such as accelerometers, gyroscopes, etc., which would not be found on many devices (e.g., laptops, Nest thermostats). We refer the reader to Sec. 2 for a more detailed survey of related work and where different systems lie with respect to the above requirements.

The main contribution of this paper is a localization system that satisfies all the three requirements above. We present SpotFi, an indoor localization system that provides a median accuracy of 40 cm using standard, commodity off-the-shelf WiFi radios, which is comparable to the best performing systems such as ArrayTrack [1] and Ubicarse [2]. SpotFi requires no hardware or baseband firmware changes/additions at the APs, nor does it need any calibration or fingerprinting of the environment. SpotFi's localization targets also have minimal requirements, they only require a commodity WiFi chip.

SpotFi incorporates three techniques that enable it to achieve this combination of accuracy and deployment simplicity:

- **Super-resolution AoA Estimation**: The first component of SpotFi is a super-resolution AoA estimation algorithm, i.e., it accurately resolves all the AoAs of the indoor multipath in spite of using just three antennas which is standard in WiFi deployments today. The number of antennas limits the number of multipath components that one can resolve. As prior work has noted [1, 2, 7, 8, 9], the more antennas there are, more multipath AoAs can be resolved more accurately. Our insight is that the multipath not only creates measurable changes in CSI across antennas because of AoA but also affects CSI across subcarriers because of time of flight (ToF, time taken by the signal to reach the AP from the localization target).

  Using this fact, instead of just estimating AoA, SpotFi combines CSI values across subcarriers and antennas to jointly estimate AoA and ToF of each path. In the process, SpotFi creates a virtual sensor array with number of elements greater than the number of multipath components, thus overcoming the constraint of limited antennas. Our unique insight here is that these joint AoA and ToF estimation algorithms can be implemented using the CSI information that is already exposed by the commodity WiFi cards. Using the AoA and ToF parameters estimated from CSI, we empirically demonstrate that, in commodity WiFi deployments, although the estimated ToF values are different from the absolute ToFs, the joint estimation procedure provides AoA accuracy that is comparable to systems that require twice as many antennas [8] or non-standard configurations such as rotating antennas used in radar [6, 2].

- **Robust Direct Path Identification**: SpotFi aims to find the AoA of the direct path component in the multipath signal from the target. However among the AoA estimates from the previous algorithm, the one corresponding to the direct path may be erroneous or may not even exist due to several practical reasons such as noisy CSI, obstructed targets, weak signal strength from the target and so on. The second key component of SpotFi is a novel algorithm that assigns values for each path depending on the likelihood that the particular path is the direct path. This metric assignment algorithm enables SpotFi to eliminate AoA estimates that are very likely to be in error and not belonging to the direct path component, thereby avoiding making large errors in localization.

- **Localization**: The final step is a localization algorithm that incorporates both the direct path AoA estimate from the above two steps as well as RSSI information available from each of the APs to calculate the location of the target. Our key contribution here is a framework that appropriately weights the information from different APs to take into account how likely it is that the AoA measurement reported by each AP corresponds to the actual direct path between the target and that AP. This weighting of information enables SpotFi to select information from APs with higher confidence metric and improve accuracy.

We implemented SpotFi using Intel 5300 commodity WiFi cards. We strived to evaluate it under the same environments as the best performing systems ArrayTrack and Ubicarse [1, 2]; we describe the testbed in Sec. 4. Our experiments show that SpotFi achieves a median localization error of 40 cm, and the 80th percentile error is 1.8 m. To put these numbers in context, the best performing prior system achieves a median error of 40 cm using 6 antennas per AP, whereas our APs have only three. We also show that SpotFi works robustly in challenging scenarios; for example, it achieves very good accuracy even when the target has strong direct paths to only a couple of APs. Finally, we also show that SpotFi is lightweight and requires CSI measurements for only ten packets from the target to localize accurately.

2. RELATED WORK

Indoor localization using wireless infrastructure is a well-studied problem. There is a large body of research which, for conciseness, we classify into four types: RSSI based, fingerprinting based, antenna array based, and time based techniques.

RSSI based approaches: This class of systems measures the RSSI from the target at multiple APs, combines them via triangulation along with a propagation model to locate the target [3, 10, 4, 5, 11, 12, 13]. These systems are easy to deploy as RSSI values are easily available in current APs. However the best known systems in this space tend to achieve a median accuracy of around 2–4 m [3, 4], and the 80th percentile error is often as high as 5 m due to insufficient modeling of RSSI which in practice depends not only on the location of the target but also on the changing environment.

Fingerprinting based approaches: The idea of these approaches is that one would collect fingerprints such as the vector of RSSIs from a specific location to all the APs in range, and if in the future a target were at the same location, it would exhibit the same fingerprint, enabling us to localize [14, 15, 16, 17, 18, 19, 20, 21, 22]. The best known such systems provide around 0.6 m of median accuracy [14], and tail accuracy on the order of 1.3 m. However these systems are difficult to deploy since they require an expensive and recurring fingerprinting operation any time there are changes in the environment (e.g., when the furniture is moved).

AoA based approaches: With the proliferation of WiFi APs with multiple antennas to support MIMO communications, antenna array based techniques which use multiple antennas at the AP have gained interest recently. The basic idea of these systems is to calculate the AoAs of the multipath signals received at each AP, find the AoA of the direct path to the target, and then apply triangulation to localize [1, 2, 6, 23, 24, 8, 25]. The best known such systems have demonstrated significant improvements in accuracy with median accuracy on the order of 0.4 m [1, 2]. However these systems are relatively difficult to deploy, since they require hardware changes by introducing as high as 8 antennas [1, 8], or new boxes itself with rotating antennas [6], or require special APs to access IQ samples [1, 24].

There are other recently proposed systems that use sensors such as gyroscopes, accelerometers, etc., along with AoA
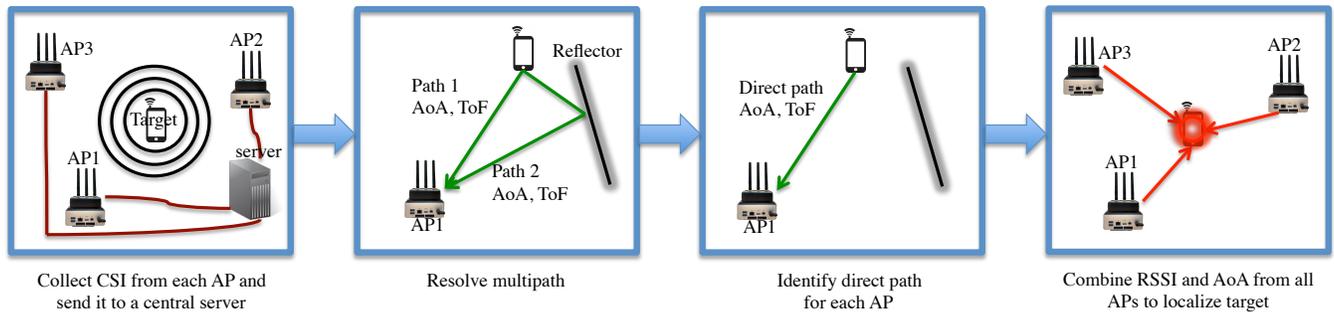
Figure 1: Overall architecture: SpotFi collects CSI and RSSI measurements from all the APs that can hear the packet transmitted by the target. In the first step, SpotFi calculates the ToF and AoA of all the propagation paths from the target to each of the APs. SpotFi then identifies the direct path between the target and the AP that did not undergo any reflections. In the final step, SpotFi estimates the location of the target by using the direct path AoA estimates and RSSI measurements from all the APs.

information [2, 23]. While these sensors are available on some phones, their accuracy varies widely across different phone models. Further there are a large number of devices that may only have a WiFi/Bluetooth chip but none of these other sensors, e.g., wearables or IoT devices such as Nest, wireless cameras, and suitcases [26, 27, 28], that would like to avail of localization capability if possible. Moreover, Ubicarse [2] requires that the human holding the device perform a specific circular motion to enable localization which is not feasible in many scenarios where localization is needed (e.g., locating misplaced or lost objects).

Time based approaches: Systems that use timestamps reported by WiFi cards can obtain time of flight at a granularity of several nanoseconds, resulting in ranging error of few meters [29, 30, 31, 32, 33, 34]. In spite of hardware/firmware modifications to overcome coarse ToF estimates, the best known systems achieve localization error of 2 m [32, 33, 30]. Some systems applied super-resolution algorithms to obtain finer ToF estimates, but require all the APs to be time synchronized [35, 36, 37, 38, 39], which is hard to achieve using commodity WiFi [40]. Algorithms for joint estimation of AoA and ToF, to improve the accuracy of both the parameter estimates, have been developed [41, 42, 43, 44, 45], and tested in simulation [46, 47, 48, 49, 50]. But these algorithms have been implemented in systems where the transmitter and receiver radios are time synchronized [51, 52], which is not possible in commodity WiFi deployments.

Other approaches: For completeness, we refer to localization systems that deploy other modalities like RFIDs [53, 54, 55], ultrawideband [56], ultrasound [57, 58, 59, 60, 61], IR [62, 63, 64], visible light [65, 66] or beacons like the ones Apple recently introduced [67]. However, we believe that none of these are ever likely to be as ubiquitous as commodity WiFi AP infrastructure.

3. DESIGN

SpotFi works in three steps:

1. Estimate the angle of arrival (AoA) and time of flight (ToF) of different multipath components of a target's signal arriving at the AP by using the CSI information that is exposed by commodity WiFi APs.
2. Estimate the likelihood that each AoA and ToF pair is the

one corresponding to the direct path between the AP and the target without any reflections.
3. Use the above information to calculate the most likely location of the target that could have produced the observed RSSI and estimated AoA.

Before we describe each step of the design in detail, we briefly discuss SpotFi's architecture. Fig. 1 shows how SpotFi would be deployed. A central server collects CSI measurements for each packet received at the APs. All the major WiFi chip families (Broadcom, Atheros, Intel, and Marvell) expose quantized CSI per subcarrier per antenna [23]. We use Intel 5300 WiFi chips in the current prototype because of the availability of CSI extraction software for these chips [68], but SpotFi can easily be deployed with WiFi APs that use chips from other manufacturers. SpotFi only adds the software required to read the reported CSI values, timestamps, and MAC addresses at the AP and ships it to the central server and nothing else. Hence we believe that SpotFi can be added to any existing, deployed WiFi AP. Also, like many other state-of-the-art localization systems, we assume that we know the locations of the APs themselves from offline, one-time measurements [1, 2, 23].

3.1 SpotFi's super-resolution algorithm for estimating AoA and ToF

A target's signal could reflect off multiple objects and arrive at the AP; typically in an indoor environment there are around 6-8 significant reflectors [69, 30, 70]. So the key question is, how might one be able to disentangle these multipath components and accurately estimate the AoA of each path even when the AP has only three antennas? To gain insight into how SpotFi solves the problem, it is helpful to understand how standard AoA computation with the well known MUSIC algorithm [7] works, which we review next.

3.1.1 Estimating AoA with MUSIC

The basic idea is that different propagation paths have different AoAs, and when the signal from a propagation path is received across an array of antennas, then the AoA will introduce a corresponding phase shift across the antennas in the array. The introduced phase shift is a function of both the distance between antennas and the AoA. To understand how MUSIC takes advantage of this fact to determine AoA,
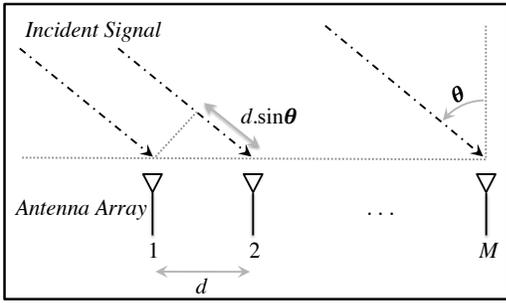
Figure 2: Uniform linear array consisting of M antennas: For AoA of θ, the target's signal travels an additional distance of $d \times \sin(\theta)$ to the second antenna in the array compared to the first antenna. This results in an additional phase of $-2\pi \times d \times \sin(\theta) \times f/c$ at the second antenna.

let's say there are L propagation paths. Let's assume that M antennas are arranged in a uniform linear array, similar to systems like ArrayTrack [1, 23], with equal spacing of d between consecutive antennas. For the kth propagation path, let θ_k denote the angle at which the signal is arriving with respect to the normal of the antenna array of the AP. Let γ_k denote the complex attenuation experienced at the first antenna in the array by the signal traveling along k^{th} propagation path. The attenuation at the second antenna in the array is the same except for an additional phase shift accumulated due to additional distance traveled by the signal, and depends on d and θ_k.

Each propagation path, therefore, has two parameters associated in this model: the attenuation and the AoA. As described in Sec. 2 of ArrayTrack [1] and illustrated in Fig. 2, relative to the first antenna in the array, the phase shift introduced at m^{th} antenna is $-2 \times \pi \times d \times (m-1) \times \sin(\theta_k) \times f/c$ where c is the speed of light and f is the frequency of the transmitted signal. For simplicity of representation, let us denote the complex exponential of these introduced phase shifts as a function of the AoA of the propagation path,

$$\Phi(\theta_k) = e^{-j2\pi \times d \times \sin(\theta_k) \times f/c}. \qquad (1)$$

So the AoA can be thought of as introducing a vector of phase shifts at the antenna array (or the sensor array). The resulting vector of received signals due to k^{th} path can be written as $\vec{a}(\theta_k)\gamma_k$, where γ_k is the complex attenuation along the path experienced at the first antenna in the array and

$$\vec{a}(\theta_k) = [1 \ \Phi(\theta_k) \ \ldots \ \Phi(\theta_k)^{(M-1)}]^{\top} \qquad (2)$$

using the notation introduced in Eq. 1. This vector $\vec{a}(\theta_k)$ is also known as the steering vector. We have as many steering vectors as propagation paths and the overall steering matrix $\mathbf{A}$ is defined as $\mathbf{A} = [\vec{a}(\theta_1), \ \ldots, \ \vec{a}(\theta_L)]$, and has dimensions $M \times L$.

The received signal vector $\vec{x}$ at the antenna array is obtained by superposition of signals due to all the paths, i.e.,

$$\vec{x} = \mathbf{A}\vec{\Gamma}, \qquad (3)$$

where $\vec{\Gamma} = [\gamma_1 \ \ldots \ \gamma_L]^{\top}$ is the vector of complex attenuations along L paths and $\mathbf{A}$ is the steering matrix. In OFDM,

data is transmitted over multiple subcarriers each with different frequency. We can write an equation similar to Eq. 3 for each of the subcarriers, and the steering matrix $\mathbf{A}$ does not change because the steering vectors do not change across closely spaced subcarriers [7, 8]. Let us construct measurement matrix $\mathbf{X}$ by using received signal vectors for each of the N subcarriers using Eq. 4 below,

$$\mathbf{X} = [\vec{x}_1 \ \ldots \ \vec{x}_N] = \mathbf{A}[\vec{\Gamma}_1 \ \ldots \ \vec{\Gamma}_N] = \mathbf{AF}, \qquad (4)$$

where vectors $\vec{x}_1, \ \ldots, \ \vec{x}_N$ denote the received signal vectors at each of the subcarriers (correspond to $\vec{x}$ in Eq. 3 obtained for different subcarriers), $\vec{\Gamma}_1, \ \ldots, \ \vec{\Gamma}_N$ are the vectors of complex attenuations of the propagation paths at each of the subcarriers (correspond to $\vec{\Gamma}$ in Eq. 3 obtained for different subcarriers), and $\mathbf{F}$ is the matrix of complex attenuations/gains.

The overall attenuation and phase shift introduced by the channel/environment measured at each subcarrier of each antenna is reported by the WiFi card as a CSI value (it's a complex number). For example, Eq. 5 represents a nominal CSI matrix reported by Intel 5300 WiFi card for 3 antennas and 30 subcarriers,

$$\text{CSI matrix} = \begin{bmatrix} \text{csi}_{1,1} & \text{csi}_{1,2} & \ldots & \text{csi}_{1,30} \\ \text{csi}_{2,1} & \text{csi}_{2,2} & \ldots & \text{csi}_{2,30} \\ \text{csi}_{3,1} & \text{csi}_{3,2} & \ldots & \text{csi}_{3,30} \end{bmatrix}, \qquad (5)$$

where $\text{csi}_{m,n}$ is the CSI value for m^{th} antenna and n^{th} subcarrier. Thus, the CSI value at each subcarrier is nothing but the received signal due to all the paths. Hence, for the AoA model described above, the received signal vector $\vec{x}$ in Eq. 3 corresponds to one of the columns in the CSI matrix and the measurement matrix $\mathbf{X}$ corresponds to the CSI matrix itself.

The MUSIC algorithm is working with this information and relationship; the WiFi card measurements provide us the matrix $\mathbf{X}$ and the goal is to estimate the matrix $\mathbf{A}$, from which it is easy to deduce the AoAs. The key idea behind the MUSIC algorithm is that *the eigenvectors of $\mathbf{XX}^{\text{H}}$ corresponding to the eigenvalue zero, if they exist, are orthogonal to the steering vectors in $\mathbf{A}$*[1]. We omit the mathematical derivation for brevity but refer to the broad literature discussing these ideas [7]. The MUSIC algorithm at a basic level proceeds first by computing the eigenvectors of $\mathbf{XX}^{\text{H}}$ corresponding to the eigenvalue zero, and then computing the steering vectors orthogonal to these vectors. Once the steering vectors are found, the AoAs can be deduced easily.

The key problem however is the assumption that there are eigenvectors of $\mathbf{XX}^{\text{H}}$ that correspond to eigenvalue zero and that they are orthogonal to the steering vectors. Prior work has shown that it's true only if the steering matrix $\mathbf{A}$ is skinny and full rank matrix and the matrix $\mathbf{F}$ is fat and full rank matrix [7]. In other words the number of rows of steering matrix $\mathbf{A}$ should be larger than the number of columns in the matrix. Physically this maps to saying that the number of sensors (in this case antennas) has to be larger than the number of propagation paths. For example, if we have radios with only three antennas and environment where there

---

[1]$\mathbf{X}^{\text{H}}$ is conjugate transpose of $\mathbf{X}$.

are more than three significant propagation paths (which is quite likely), the above algorithm doesn't work well. This is the reason past works have either used more antennas (eight in ArrayTrack [1]) or used rotating antennas to simulate a larger antenna array (e.g., LTEye [6]). Also, the number of columns of matrix $\mathbf{F}$ of complex gains should be greater than the number of its rows, i.e., number of measurements at the sensor array should be greater than the number of paths.

3.1.2 Super-Resolution AoA Estimation

How might one increase the resolution of AoA estimation? From the above discussion it's clear that the key factor is the number of sensors from which we can measure properties of the propagation paths and the number of independent measurements we can obtain at the sensor array. SpotFi's insight is that the number of sensors is not limited by the number of antennas, but in fact by leveraging the fact that WiFi has many OFDM subcarriers on each of which we get a CSI measurement, the number of sensors can be expanded to be equal to the product of the number of subcarriers and the number of antennas. For example, for Intel 5300 WiFi cards, number of sensors would be equal to $30 \times 3 = 90$ sensors rather than the 3 antennas with the modeling described in Sec. 3.1.1.

However, if each sensor in this extended sensor array is modeled to measure just AoAs of the paths, then the number of distinct values in our parametric model of sensor measurements is still limited by the number of antennas. This is because AoA of a propagation path does not manifest itself in any measurable way across subcarriers, i.e., AoA does not introduce any phase shift across subcarriers of an antenna. This is easy to see given that the relative phase shift introduced by AoA of the kth path across two subcarriers of the mth antenna is $2\pi(m-1)d(f_i - f_j)\sin(\theta_k)/c$, where f_i and f_j denote the frequencies of the two subcarriers. So, with half-wavelength antenna spacing, for two subcarriers of the second antenna separated by 40 MHz, any AoA introduces a phase shift of at most 0.002 radians which is negligible. Hence, phase shift due to AoA is same across all the subcarriers of an antenna since the speed of light factor in the denominator is much larger than this small frequency difference. So, although there are 90 subcarriers (from all three antennas combined) in Intel 5300 CSI measurements, only 2 distinct phase shifts are introduced due to AoA, since we compute phase shifts relative to the first antenna and there are three antennas.

Obtaining sensor array larger than the number of paths: SpotFi's key insight is counterintuitive. Instead of just estimating AoA per propagation path, SpotFi proposes to also calculate the time of flight (ToF) for each path. By definition each path will likely have a different ToF too. The reason to add ToF as a parameter for each path is that it introduces measurable phase shifts across subcarriers. For example the phase shift across two subcarriers even across the same antenna for the kth path with ToF τ_k is given by $2\pi(f_i - f_j)\tau_k$ which is significant (the reason numerically is the lack of the speed of light factor in the denominator). Here, f_i and f_j denote the frequencies of the two subcarriers as before. For

example, for two subcarriers spaced apart by 40 MHz and for ToF of 10 ns, there is significant difference of 2.5 radians in the phase shift introduced at the two subcarriers. Specifically, for equispaced OFDM subcarriers, k^{th} path with ToF τ_k introduces a phase shift of $-2 \times \pi \times (n-1) \times f_\delta \times \tau_k$ at the nth subcarrier relative to the first subcarrier of an antenna, where f_δ is the frequency spacing between two consecutive subcarriers. For simplicity of representation, let us denote the complex exponential of the phase shift introduced between two adjacent subcarriers as a function of the ToF,

$$\Omega(\tau_k) = \mathrm{e}^{-j2\times\pi\times f_\delta\times\tau_k}. \tag{6}$$

SpotFi exploits this insight to expand the number of sensors, and specifically design a steering matrix $\mathbf{A}$ that is skinny and enables the resolution of all the paths.

Specifically, consider the sensor array comprising of all the subcarriers at all the antennas. The measurement matrix $\mathbf{X}$ is constructed by stacking CSI from all the subcarriers at all the antennas, and hence is a single column matrix. Each propagation path introduces a distinct phase shift at each of the sensors depending both on its ToF and AoA. So, for a path with AoA θ and ToF τ, the steering vector, for the $M \times N$ sensors (M antennas times N subcarriers), is formed by phase shifts introduced at each of the sensors due to both AoA and ToF, and is given by:

$$\vec{a}(\theta,\tau) = [\overbrace{1,\ldots,\Omega_\tau^{N-1}}^{\text{antenna 1}}, \underbrace{\Phi_\theta,\ldots,\Omega_\tau^{N-1}\Phi_\theta}_{\text{antenna 2}},\ldots, \overbrace{\Phi_\theta^{M-1},\ldots,\Omega_\tau^{N-1}\Phi_\theta^{M-1}}^{\text{antenna M}}]^\top,$$
$$\tag{7}$$

where $\Omega(\tau)$ is written as Ω_τ and $\Phi(\theta)$ as Φ_θ for brevity.

CSI at each sensor is a linear combination of the phase shifts introduced due to all the paths weighted by their attenuations. So, the newly constructed measurement matrix $\mathbf{X}$ is nothing but a linear combination of the steering vectors in Eq. 7 evaluated for all the paths. Note that phase of complex attenuation in the model described in Eq. 3 has the phase shift due to ToF absorbed into it and is different for different subcarriers, whereas here the phase of attenuation just depends on the objects that the signal interacted with along the path and is same for all the subcarriers of all the antennas. The steering matrix corresponding to this extended sensor array still has number of columns equal to the number of paths. We have thus increased the number of sensors without increasing the number of paths, i.e., we have achieved a skinny steering matrix $\mathbf{A}$.

The other fact to note from the previous section's description of MUSIC is that the number of measurements at the sensor array, that can be written as linear combination of the *same steering vectors*, should be greater than the number of paths. However, the measurement matrix obtained by stacking CSI from all the subcarriers at all the antennas is a single column unit rank matrix. We now describe how we obtain a measurement matrix with number of columns greater than number of paths.

CSI smoothing: SpotFi's mathematical trick to obtain a sensor array with multiple independent measurements is best demonstrated through an example. Let's say there are $L = 2$

5 subcarriers

$$
\begin{array}{ccccc}
1 & \Omega_{\tau_k} & \Omega^2_{\tau_k} & \Omega^3_{\tau_k} & \Omega^4_{\tau_k} \\
\Phi_{\theta_k} & \Omega_{\tau_k}\Phi_{\theta_k} & \Omega^2_{\tau_k}\Phi_{\theta_k} & \Omega^3_{\tau_k}\Phi_{\theta_k} & \Omega^4_{\tau_k}\Phi_{\theta_k} \\
\Phi^2_{\theta_k} & \Omega_{\tau_k}\Phi^2_{\theta_k} & \Omega^2_{\tau_k}\Phi^2_{\theta_k} & \Omega^3_{\tau_k}\Phi^2_{\theta_k} & \Omega^4_{\tau_k}\Phi^2_{\theta_k}
\end{array}
$$

3 antennas

(a) Entries of the steering vector

$$
\begin{bmatrix}
\mathrm{csi}_{1,1} & \mathrm{csi}_{2,3} \\
\mathrm{csi}_{1,2} & \mathrm{csi}_{2,4} \\
\mathrm{csi}_{1,3} & \mathrm{csi}_{2,5} \\
\mathrm{csi}_{2,1} & \mathrm{csi}_{3,3} \\
\mathrm{csi}_{2,2} & \mathrm{csi}_{3,4} \\
\mathrm{csi}_{2,3} & \mathrm{csi}_{3,5}
\end{bmatrix}
=
\begin{bmatrix}
1 & 1 \\
\Omega_{\tau_1} & \Omega_{\tau_2} \\
\Omega^2_{\tau_1} & \Omega^2_{\tau_2} \\
\Phi_{\theta_1} & \Phi_{\theta_2} \\
\Omega_{\tau_1}\Phi_{\theta_1} & \Omega_{\tau_2}\Phi_{\theta_2} \\
\Omega^2_{\tau_1}\Phi_{\theta_1} & \Omega^2_{\tau_2}\Phi_{\theta_2}
\end{bmatrix}
\begin{bmatrix}
\alpha_1 & \alpha_1\Omega^2_{\tau_1}\Phi_{\theta_1} \\
\alpha_2 & \alpha_2\Omega^2_{\tau_2}\Phi_{\theta_2}
\end{bmatrix}
$$

CSI values at the two subarrays   **Same steering matrix**   **Independent vectors of gains**

(b) CSI values at the two sensor subarrays

Figure 3: (a) The elements of the steering vector in Eq. 7 for two subsets of sensors. The elements in the dashed red box (second sensor subarray) are obtained by scaling the corresponding elements in solid blue box (first subarray) by $\Omega^2(\tau_k)\Phi(\theta_k)$. Ω_{τ_k} represents $\Omega(\tau_k)$ and Φ_{θ_k} represents $\Phi(\theta_k)$. (b) $\mathrm{csi}_{m,n}$ represents the CSI obtained at n^{th} subcarrier and m^{th} antenna. So, the two columns on the left hand side of the equation correspond to the CSI values obtained at the two sensor subarrays displayed in (a). CSI measurements obtained for the second sensor subarray are obtained by combining the same steering vectors as the first sensor subarray but with different independent vector of weights. The CSI measurements and the vector of gains corresponding to the first sensor subarray are colored blue and similarly the corresponding values for second sensor subarray are colored red.

paths. Consider the CSI at the subcarriers corresponding to two sensor subarrays displayed in the two columns of the matrix on the left hand side (LHS) of equation in Fig. 3(b). First sensor subarray, corresponding to the first column of Fig. 3(b), is comprised of first three subcarriers at antennas 1 and 2. The second sensor subarray, corresponding to the second column of Fig. 3(b), is comprised of subcarriers 3 to 5 at antennas 2 and 3. We will show in the following paragraphs that the measurements at these two sensor subarrays can be written as linear combination of the same vectors, but with different gains, as illustrated in Fig. 3(b). So, if we can identify L such sensor subarrays whose CSI values can be written as linear combination of same vectors, then the MUSIC algorithm can be applied on the measurement matrix obtained from the CSI measurements at these subarrays.

To gain intuition into why CSI at the two sensor subarrays shown in Fig. 3(b) can be written as linear combination of same vectors, let's look at the entries of steering vector in Eq. 7 evaluated for k^{th} propagation path as illustrated in Fig. 3(a). The phase shifts in the solid blue box correspond to the steering vector entries of the first sensor subarray and the phase shifts in the dashed red box correspond to those

in the second sensor subarray. We observe that the phase shift between the corresponding elements of the two sensor subarrays are related through a common scaling factor. For example, phase shift of the top left sensor of second sensor subarray, i.e., subcarrier 3 at antenna 2, is obtained by multiplying phase shift at the top left sensor of the first sensor subarray, i.e., subcarrier 1 at antenna 1, by $\Omega^2(\tau_k)\Phi(\theta_k)$. This scaling factor is infact the phase shift term we would see for this propagation path for a shift of 2 subcarriers and 1 antenna, and is expected since both the sensor subarrays are structurally same except for a shift of 2 subcarriers and 1 antenna. The common scaling factor, $\Omega^2(\tau_k)\Phi(\theta_k)$, depends on the relative shift in antennas and subcarriers of the two sensor subarrays as well as the propagation path parameters, and hence is different for different paths. How can one exploit the above insight?

Let α_1 and α_2 be the complex gains along the two paths. We obtain CSI values at the sensors belonging to first subarray by weighing the corresponding steering vector entries for the two paths by their complex gains α_1 and α_2 (see Fig. 3(b)), and similarly for the second sensor subarray. However, by absorbing common scaling factor $\Omega^2(\tau_k)\Phi(\theta_k)$ into the complex gains at the second subarray, we can write the CSI values at the second sensor subarray by weighing the steering vector entries corresponding to first subarray for the two paths with the modified gains $\Omega^2(\tau_1)\Phi(\theta_1)\alpha_1$ and $\Omega^2(\tau_2)\Phi(\theta_2)\alpha_2$. The same is illustrated in Fig. 3(b). We have thus shown that CSI at these two sensor subarrays can be written as linear combination of the same vectors. Moreover, the vector of complex gains of the second subarray is linearly independent of the vector of the complex gains of first subarray, because attenuations corresponding to different paths are multiplied by different factors to obtain modified gains at the second subarray.

By generalizing these observations, one can prove that the CSI values at different sensor subarrays, that are obtained by shifting a fixed subset of sensors, can be written as linear combination of the same vectors. Infact, the steering vector entries of the fixed subset of sensors for different paths now form the steering matrix **A** and by combining the vectors of this steering matrix with different weights, we obtain CSI values at different sensor subarrays. So, if we construct a measurement matrix **X** using CSI values at different sensor subarrays that are structurally similar but shifted from each other, we can successfully apply MUSIC using this new measurement matrix **X**. Specifically, let's consider CSI measurements from Intel 5300 cards. SpotFi considers CSI values at sensor subarrays that are formed by different shifts of the fixed sensor subarray comprised of the first 15 subcarriers of first two antennas to construct the measurement matrix as illustrated in Fig. 4. We term this new measurement matrix **X** as *smoothed CSI* matrix. In this construction, steering matrix **A** is of dimensions $30 \times L$, where 30 rows is from the fact that each sensor subarray consists of the combinations of 15 subcarriers at 2 antennas and L is the number of propagation paths. The matrix **F**, due to the shifting trick, now has dimensions $L \times 30$, where 30 columns is from the fact that 30 is the maximum number of sensor subarrays we

$$\begin{bmatrix} \text{csi}_{1,1} \\ \vdots \\ \text{csi}_{1,30} \\ \text{csi}_{2,1} \\ \vdots \\ \text{csi}_{2,30} \\ \text{csi}_{3,1} \\ \vdots \\ \text{csi}_{3,30} \end{bmatrix} \Rightarrow \begin{bmatrix} \text{csi}_{1,1} & \text{csi}_{1,2} & \cdots & \text{csi}_{1,16} & \text{csi}_{2,1} & \cdots & \text{csi}_{2,16} \\ \text{csi}_{1,2} & \text{csi}_{1,3} & \cdots & \text{csi}_{1,17} & \text{csi}_{2,2} & \cdots & \text{csi}_{2,17} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \text{csi}_{1,15} & \text{csi}_{1,16} & \cdots & \text{csi}_{1,30} & \text{csi}_{2,15} & \cdots & \text{csi}_{2,30} \\ \text{csi}_{2,1} & \text{csi}_{2,2} & \cdots & \text{csi}_{2,16} & \text{csi}_{3,1} & \cdots & \text{csi}_{3,16} \\ \text{csi}_{2,2} & \text{csi}_{2,3} & \cdots & \text{csi}_{2,17} & \text{csi}_{3,2} & \cdots & \text{csi}_{3,17} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \text{csi}_{2,15} & \text{csi}_{2,16} & \cdots & \text{csi}_{2,30} & \text{csi}_{3,15} & \cdots & \text{csi}_{3,30} \end{bmatrix}$$

$$\textbf{CSI}_{90\text{x}1} \qquad\qquad \textbf{Smoothed CSI}_{30\text{x}30}$$
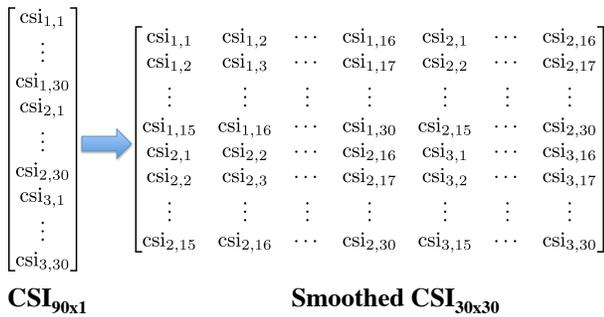
Figure 4: This figure illustrates construction of smoothed CSI matrix from the input CSI measurements. $\text{csi}_{m,n}$ is the CSI value for m^{th} antenna and n^{th} subcarrier. MUSIC algorithm can be directly applied on the smoothed CSI matrix to obtain the AoA and ToF of all the propagation paths.

can form by trying all possible shifts of 15 subcarriers at 2 antennas in a (30 subcarriers × 3 antennas) system.

At this stage, the standard MUSIC algorithm can be applied on smoothed CSI matrix to find the steering vectors and then the AoA and ToF parameters corresponding to each path. The algorithm now works well because by using the smoothed CSI matrix, SpotFi manages to ensure that the number of sensors is greater than the number of paths as SpotFi now has 30 sensors compared to 3 before.

This method of using shifted subsets of sensors is similar to the spatial smoothing technique [9] applied in localization systems like ArrayTrack [1], where the idea is to use measurements from different antenna subarrays formed by considering different subsets of antennas. In SpotFi, we consider different subsets of the antennas and subcarriers together. Algorithms to jointly estimate AoA and ToF by using different sensor subarrays have been explored in literature [42, 43], and applied in systems where all the WiFi radios are time synchronized [51, 52]. However, SpotFi is the first system that applies these algorithms with commodity WiFi deployments where the APs and the targets are not time synchronized. The idea is that although different WiFi cards are not time synchronized, all the transceiver chains on a single WiFi card share the same sampling clock. As a result, lack of time synchronization introduces the same phase shift in the CSI observed at different antennas of an AP which we describe later in Sec. 3.2.2. This phase shift corrupts the ToF values but the AoA values stay the same as in the case where the radios of the AP and the target are time synchronized. Thus the joint AoA and ToF estimation algorithms are still useful in achieving improved AoA accuracy in spite of having just three antennas at the AP which we empirically demonstrate in Sec. 4.

3.2 Identifying Direct Path AoA

SpotFi's next step is to determine if direct path exists and identify the AoA of the direct path from the target to the AP. As described in section 3.1, SpotFi's super-resolution algorithm provides ToF and AoA for all the paths from the target to an AP. A natural heuristic is to exploit the ToF information. Specifically, at each AP we can identify the path with the lowest ToF and declare that as the direct path. Further, we can use the ToF value along with the speed of light to compute the distance of the target from the AP. In theory, this information at a single AP along with AoA information would be sufficient to localize the target.

The above reasonable technique however does not work in practice because the ToF estimates do not capture the true time taken by the signal to travel from the target to the AP. The reason is that in standard WiFi networks, the sender and the receiver are not time synchronized; so their sampling clocks at the DAC and the ADC are not in sync. Hence the ToF estimate also includes delays from sampling time offset (STO) between the target and AP and is therefore not a true measure of the distance traveled.

However, since the ToFs of all the paths will have the same delay added to them due to STO, we can still use the path with the lowest ToF to identify which path traveled the shortest relative distance. But the path with the smallest estimated ToF may not be the direct path. In many indoor scenarios, the direct path may be too weak or nonexistent due to obstruction by walls, etc. In these scenarios, SpotFi's super-resolution algorithm will not even identify the direct path and all the ToF/AoA estimates will be for indirect paths. In this case assuming that the path with the lowest ToF is the direct path would be a mistake. But in cases where there is a strong direct path, the heuristic works and can be used to calculate the AoA of the direct path. How might one distinguish between these two cases, when using lowest ToF heuristic, without knowing the ground truth?

3.2.1 Using AoA and ToF from Multiple Packets

We build on a practical observation that was articulated in earlier localization systems [1, 24]: the AoA (and even ToF) estimates of the direct path over a few consecutive packets will show much smaller variation compared to the estimated AoAs (and ToFs) of indirect paths. We demonstrate this observation in Fig. 5(c) through AoA and ToF estimates calculated from CSI traces collected for one of our experiments. SpotFi leverages this observation to estimate likelihoods for each of the paths to be the direct path between the target and the AP. The idea is to form a measure of the variations of the AoAs and ToFs of paths estimated over consecutive packets, and assign a likelihood metric for each of the paths that is inversely proportional to the amount of variation in the AoA and ToF estimates for that path.

Although measuring variation in the AoA parameter alone is easy, measuring the variation in the ToF parameters is challenging. Apart from sampling time offset, there is also a sampling frequency offset (SFO) between every WiFi sender-receiver pair. SFO changes the sampling time offset from packet to packet for the same sender-receiver pair, which in turn results in additive noise to the ToF estimates across packets. So the variance of ToFs estimated by SpotFi's super-resolution algorithm includes this additional variance introduced due to varying sampling time offset. So for every packet's measurements, we need to remove the effect of the random sampling time offset before estimating variance across packets for AoA and ToF estimates.

3.2.2 Sanitizing ToF Estimates

STO adds a constant offset to the ToF estimates of all the paths. This common additional delay manifests itself as a linear in frequency term in the phase response of the channel. Hence, STO of τ_s results in adding $-2\pi f_\delta (n-1)\tau_s$ to the phase of CSI value of the nth subcarrier. The additional phase induced by STO is the same across antennas for a particular subcarrier, as all the receiver chains of the same WiFi card are time synchronized. We will now show that removing the linear fit that is common to the unwrapped phase response of all the antennas before estimating multipath parameters eliminates the varaince due to changing STO.

Let us consider two consecutive packets transmitted by the target. Let $\psi_i(m,n)$ represent unwrapped phase response of the channel for the ith packet at n^{th} subcarrier of mth antenna, and $\tau_{s,i}$ is the STO for i^{th} packet. By applying ToF sanitization algorithm described in Algorithm 1, say we removed the linear fit of CSI phase response for first packet to obtain modified phase response $\widehat{\psi_1}(m,n)$.

Algorithm 1: SpotFi's ToF sanitization algorithm

Data: Unwrapped CSI phase ψ_i for i^{th} packet

1 Obtain the best linear fit of the unwrapped CSI phase as

$$\widehat{\tau_{s,i}} = \arg\min_{\rho} \sum_{m,n=1}^{M,N} \left(\psi_i(m,n) + 2\pi f_\delta(n-1)\rho + \beta \right)^2 ;$$

2 From the unwrapped CSI phase, subtract the phase that would have been added due to STO $\widehat{\tau_{s,i}}$ to obtain modified CSI phase $\widehat{\psi_i}(m,n)$ as

$$\widehat{\psi_i}(m,n) = \psi_i(m,n) + 2\pi f_\delta(n-1)\widehat{\tau_{s,i}}$$

Phase response of second packet ψ_2 can be written as $\psi_2(m,n) = \psi_1(m,n) - 2\pi f_\delta(n-1)(\tau_{s,2} - \tau_{s,1})$. Using this relationship, one can prove that the modified CSI phase of the second packet is given by $\widehat{\psi_2}(m,n) = \psi_1(m,n) + 2\pi f_\delta(n-1)\widehat{\tau_{s,1}}$, which is same as the modified CSI phase of first packet. The actual and modified CSI phase responses for two packets obtained from CSI traces collected from our experiments are presented in Fig. 5(a) and Fig. 5(b) respectively. The modified CSI phase response, obtained by applying Algorithm 1 for each packet, does not change even if the STO changes, and hence is free from the variations of STO. So, the ToF parameters estimated across packets using modified CSI are free from variance of changing STO. We note that the Algorithm 1 is similar to the data sanitization process in PinLoc [15] and is an extension of the process to multiple antennas. We also note that although we have discussed changes in ToF due to SFO alone, the variance in ToF due to random packet detection delay [40] can also be eliminated by following Algorithm 1.

3.2.3 Estimating Direct Path Likelihoods

Now we have a collection of AoA and ToF estimates whose variance across packets can be estimated. To assign likelihood estimates for each of the estimated paths to be the direct path between the target and AP, we plot AoA and ToF estimates from multiple measurements in a two- dimensional (one each for AoA and ToF) space and apply a clustering algorithm. The intuition is that AoA and ToF estimates from the same path but different packets will be clustered together, but the diameter of each cluster (i.e., the tightness of each cluster) will be a function of the variations in AoA and ToF values for the corresponding path across packets.

Specifically, we use well-known Gaussian Mean clustering algorithm with five clusters to identify the clusters of the estimated parameters. The number of clusters is chosen as five because typically we see at best five significant paths in an indoor environment [1, 8, 24]. The mean of a cluster is used as an estimate for the actual ToF and AoA of the particular propagation path. The variance in the ToF of a path is estimated by calculating the population variance of the ToF estimates belonging to the cluster of the particular path, and similarly for AoA. Then the likelihood for k^{th} path to be the direct path is calculated as

$$likelihood_k = \exp(w_C \bar{C}_k - w_\theta \bar{\sigma}_{\theta_k} - w_\tau \bar{\sigma}_{\tau_k} - w_s \bar{\tau}_k), \ (8)$$

where $likelihood_k$ is the likelihood that the kth path is the direct path, $\bar{C}_k$ is the number of points in the cluster corresponding to that path, $\bar{\sigma}_{\theta_k}$ and $\bar{\sigma}_{\tau_k}$ are the population variances of the estimated AoA and ToF respectively for points belonging to that cluster, and $\bar{\tau}_k$ is the mean of ToF for points in that cluster. The weighting factors w_C, w_θ, w_τ, and w_s are constants to account for different scales of the corresponding terms (for example, ToF values are on the order of ns and number of points in the cluster is on the order of 10).

The likelihood estimate incorporates a few other terms apart from the tightness of the cluster. First is the term corresponding to the number of points in the cluster. The insight here is that if a cluster corresponds to a physical propagation path, then it is likely to have more measurements than a cluster which is spurious and doesn't correspond to an underlying physical path. The intuition for the term related to the mean ToF is that the direct path will have the smallest ToF, so a higher ToF term should signify a lower likelihood.

SpotFi declares the path with the highest likelihood metric as the direct path, and stores the AoA and the likelihood value of the corresponding path.

3.3 Localizing the Target

Next, SpotFi attempts to localize the target by combining the direct path AoA estimates and their likelihood values corresponding to different APs. Further, the server also has the access to the RSSI measurements for the packets from the target to each AP that "heard" the target. The server assumes a standard widely used path loss model to relate RSSI to distance as described in prior work [3, 71]. The server then fuses all this information to localize the target.

To localize, SpotFi finds the location that best explains the AoA and RSSI measurements at different APs. We do not use the ToF information to form distance estimates because it still does not capture the true ToF of the signal from the target to the AP. The procedure described in Sec. 3.2.2 to scrub any distortion in the ToF estimates only helps in removing

(a) Unwrapped CSI phase   (b) Modified CSI phase   (c) AoA-ToF parameters for different packets
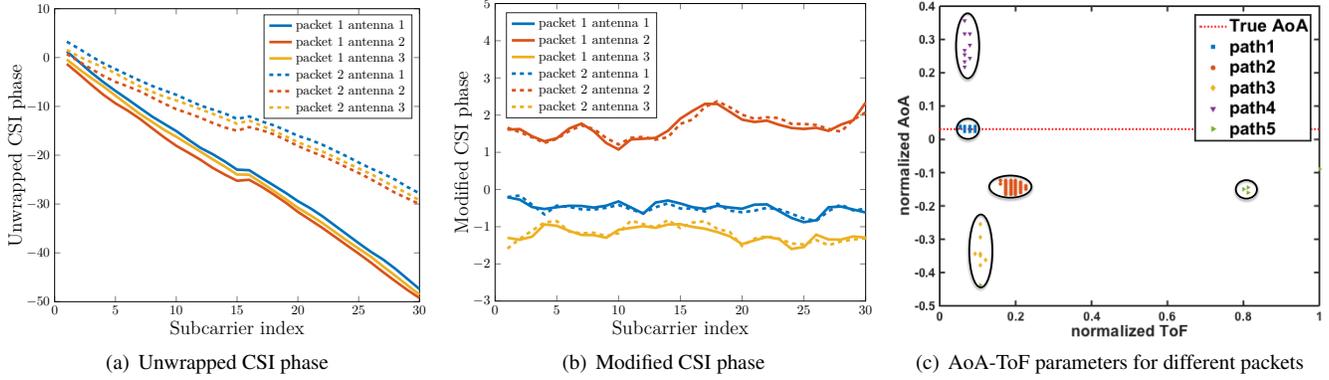
Figure 5: (a)Unwrapped CSI phase response for two packets. The added phase due to STO is the same for the CSI observed at any of the antennas. (b) Modified CSI phase response obtained after applying Algorithm 1 is same for the two packets in spite of different STOs. (c) ToF-AoA clusters obtained using modified CSI for 170 packets. ToF and AoA values are normalized so that their values lie in the same range. Both path1 and path4 have similar ToF but the reflected path, path4, has a much higher variance in its estimates than the direct path, path1. The estimates of direct path form a tight cluster compared to other paths. SpotFi's direct path likelihood metric rightly chose path1 as direct path as it considers both smallest ToF and tightness of cluster.

the effect of varying STOs and normalize the ToF values across packets to a constant STO, but does not give us the true ToF. To do so, one would need very precise (nanosecond level) synchronization across nodes or across APs. Recent work [40] has shown the feasibility of such synchronization but such mechanisms are not available in current, deployed WiFi infrastructure. Hence, SpotFi uses its ToF estimates only for likelihood estimates but not for determining range.

Specifically, for localization, SpotFi finds the location that minimizes the deviation between AoA and RSSI values that would have been observed at each AP if the target was actually at that location and the corresponding values that were actually observed at each AP. We measure the deviation using standard least squares cost. Mathematically, we find the location that minimizes the following objective function:

$$\sum_{i=1}^{R} l_i[(\bar{p}_i - p_i)^2 + (\bar{\theta}_i - \theta_i)^2], \qquad (9)$$

where there are R APs, i^{th} AP has direct path AoA θ_i and observed RSSI p_i, and $\bar{\theta}_i$ and $\bar{p}_i$ are the AoA and RSSI respectively that would have been observed at the ith AP if the target was transmitting from that location. The weighting factor l_i is the likelihood value of most likely candidate for the direct path from i^{th} AP.

The deviations for different APs have been weighted differently to realize the logical intuition that the APs which have a lower likelihood metric in their estimate of the direct path AoA should be penalized and those with higher likelihood metric should be rewarded. We cannot directly apply convex optimization techniques to find the location that minimizes the objective 9 because the RSSI and AoA observed at an AP are non-convex in terms of the location coordinates. Hence SpotFi applies a well known heuristic known as sequential convex optimization [72] to convexify the objective function piece-wise and obtains the target location that minimizes the objective function 9.

Thus SpotFi accurately determines the location of the target by effectively filtering the reflection paths through like-

lihood values and then efficiently calculating the target location that best explains the RSSI and estimated direct path AoA observed at the APs. We summarize the complete localization algorithm in Algorithm 2.

Algorithm 2: SpotFi's localization algorithm

Data: CSI and RSSI measurements from target to each of the R APs, and the locations of the APs

Result: Location of the target

1 **for** *each AP* $i \in 1, 2, \ldots, R$ **do**
2   **for** *each packet* $p \in 1, 2, \ldots, 10$ **do**
3   Remove linear fit in CSI phase response by applying Algorithm 1 ;
4   Obtain smoothed CSI matrix **X** as in Fig. 4 ;
5   Construct matrix E_N whose columns are eigenvectors of $\mathbf{XX}^{\text{H}}$ corresponding to eigenvalues that are smaller than a threshold;
6   Evaluate MUSIC spectrum $P_{MU}(\theta, \tau) = \frac{1}{(\vec{a}^{\text{H}}(\theta,\tau) E_N E_N^{\text{H}} \vec{a}(\theta,\tau))}$;
7   Obtain AoA and ToF of multipath components as peaks of MUSIC spectrum ;
8   **end**
9   Cluster AoA and ToF from multiple packets ;
10   Declare AoA of cluster with highest likelihood value (use Eq. 8) as direct path AoA θ_i ;
11 **end**
12 Minimize objective 9 with optimization variables as target's location and path loss model parameters

---

4. EXPERIMENTAL EVALUATION

4.1 Implementation

We implemented our system using off-the-shelf Intel 5300 WiFi NICs. We employed Linux CSI tool [68] to obtain the PHY layer CSI information for each packet. The WiFi NICs
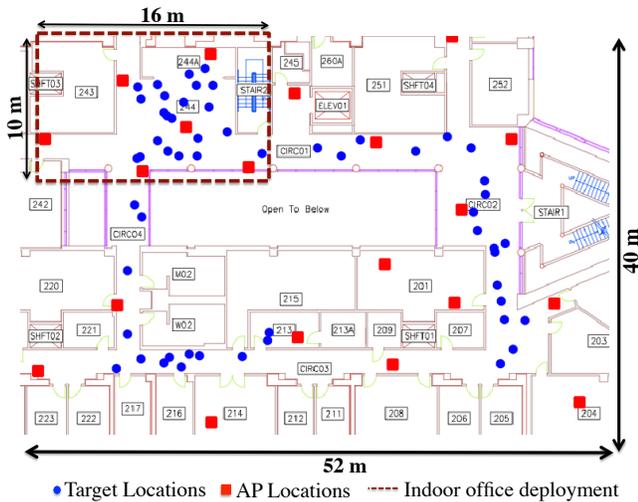
Figure 6: Experiment testbed showing the target locations (blue circles) and the AP locations (red squares). The testbed is designed to evaluate SpotFi in wide variety of deployment scenarios. The region in dashed red box, covering 16 m × 10 m area, represents typical indoor office environment. SpotFi has also been tested when both the APs and the targets are along a corridor. SpotFi has been further stress-tested in locations where the localization targets have atmost two APs in LoS.

operate in 5 GHz WiFi spectrum because of firmware limitations [8]. Note that operating in 5 GHz frequencies is more challenging compared to 2.4 GHz because of higher attenuation. The firmware provides CSI at only 30 of the sub-carriers although the data is sent on 116 subcarriers for 40 MHz bandwidth. Also, the CSI information is quantized, i.e., each of real and imaginary parts of CSI for every sub-carrier is represented using 8 bits.

The system consists of multiple computers equipped with WiFi cards which act as access points. One machine which is mobile on a cart acts as the target whose location is of interest. The locations of the access points with respect to a map are measured accurately by using laser range finder and architectural drawings of the building where we deployed SpotFi. The APs operate in monitor mode and obtain the CSI for the packets transmitted by the target. The APs export the CSI, along with a timestamp at which CSI is obtained, to a central server. The server processes and combines the CSI information from multiple access points to determine the location of the target. SpotFi's localization algorithm, that is described in Algorithm 2 and is executed on the server, is implemented using MATLAB in our current prototype and has not been optimized for speed.

Deployments tested: We deployed SpotFi in different locations of our building. Specifically, we deployed SpotFi over almost the entire floor of our building. The building has a variety of spaces, from students labs to a cafeteria to long corridors with smaller offices. Locations of WiFi APs and 55 target locations are depicted in Fig. 6.

Compared Approaches: We compare SpotFi with practical implementation of ArrayTrack based on CSI from a WiFi NIC with three antennas and no further hardware modifications [8]. Throughout this section, ArrayTrack refers to

the localization application described in Phaser [8], which is mathematically similar to the method applied in Array-Track [1]. We chose ArrayTrack with three antennas as the comparison point because among recently proposed designs it is the best performing system which can be deployed without any hardware or firmware modifications at the APs. Note that we do not use the ArrayTrack implementation described in an earlier paper [1] with WARP boards because it needed 6–8 antennas to be effective. We also do not use a recent proposal [8] from the same group to stitch together two WiFi NICs with additional hardware to create a five antenna AP since once again it requires hardware modifications. Our implementation faithfully reproduces the ArrayTrack algorithm and design but is constrained to use three antennas and the CSI measurements from commodity APs.

Systems such as Ubicarse [2] do not require infrastructure hardware/firmware modifications, but they require that the target has gyroscopes and at least two antennas, and moreover that it should be held by a human and he/she moves it in a particular circular way. SpotFi's requirements do not allow for any such assumptions, localization has to work even if the target has a simple, single antenna commodity WiFi chip and nothing else and is on a completely static target (e.g., locating a phone lost somewhere in a home).

4.2 Overview of Evaluation

We tested a series of hypotheses to understand how well SpotFi performs when deployed in typical indoor environments. Following is a summary of the main results from evaluation.

- SpotFi achieved median localization error of 0.4 m in deployments similar to those used for testing ArrayTrack and Ubicarse.
- Even in challenging conditions where the target has only two APs with a robust direct path, SpotFi still performs well with median localization error of 1.6 m compared to ArrayTrack whose median error is 3.5 m. Similarly in scenarios such as corridors where even though there may be multiple APs the target is not surrounded in all directions by them, SpotFi provides a median error of 1.1 m compared to 4 m for ArrayTrack.
- SpotFi's accuracy stems from two key components. First is its super-resolution AoA estimation algorithm whose median error is less than 5 degrees in LoS conditions and less than 10 degrees in NLoS conditions. The MUSIC algorithm used in practical implementation of ArrayTrack only achieves an estimation error of 7.4 and 15.2 degrees respectively for the same scenarios. Second, SpotFi's likelihood estimation algorithm ensures that among the available estimates, incorrect estimates for the direct path are likely not picked, and even if they are, they will be weighted with low confidence and have a lesser impact on the location estimate.

4.3 SpotFi's Accuracy

In this evaluation, we test the localization accuracy achieved by SpotFi in different deployment scenarios.

| (a) Indoor office deployment | (b) High NLoS deployment | (c) Corridors |
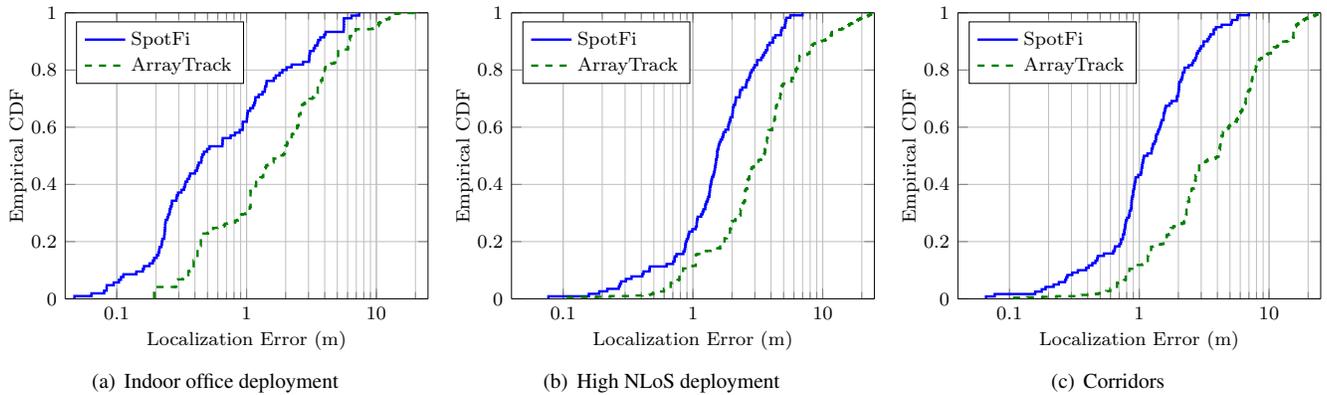
Figure 7: Plots CDFs of SpotFi's localization error for different deployments in the testbed in Fig. 6 and compares with the localization error achieved by practical implementation of ArrayTrack with three antennas for the same data.

4.3.1 Indoor Office Deployments

Localization accuracy is dependent on the multipath environment, the material used in walls, the presence of metallic objects, the density of WiFi AP deployment and many other factors. Hence we start first by replicating the deployment scenario used in evaluating state-of-the-art systems such as ArrayTrack and Ubicarse. Specifically both these systems used an indoor office environment with an area of roughly 15×10 sq.m and deployed five-six APs to span the area. Target locations are also located within the same area. The environment is very multipath rich and typically has 4–5 APs with a sufficiently strong direct path. Both ArrayTrack (see Sec. 6 and 4 in [1]) and Ubicarse (see Sec. 8 in [2]) use a very similar environment. Our testbed with APs and locations for this experiment are highlighted by a dashed red box in Fig. 6.

Method: All experiments are conducted as follows. First, all APs are set in monitor mode on a channel with 40 MHz bandwidth in the 5GHz band. For every target location shown in the testbed, we place our Intel NUC based WiFi client and configure it to be on the same channel as the APs. First we measure the ground truth location of the client using the lasers as described above. The target then transmits 500 packets with 100 ms interval and six of our AP nodes surrounding the client that can hear the client log the packets as well as the CSI and MAC address associated with that packet. The measured CSI and MAC address traces are sent with timestamps to the central server. The server chops up the CSI traces into groups of forty consecutive CSI measurements at a time on which the localization algorithm 2 is applied. The same data is also used for ArrayTrack as described above. We repeat this experiment for each target location in the testbed. Fig. 7(a) shows the CDF of localization errors for SpotFi and ArrayTrack.

Analysis: We observe from Fig. 7(a), SpotFi achieves a median localization error of 0.4 m compared to 1.8 m for ArrayTrack. The 80th percentile tail errors for SpotFi and ArrayTrack are 1.8 m and 4 m respectively. To put these numbers in context, ArrayTrack with six-eight antennas and Ubicarse/LTEye with rotating antennas achieve between 0.3–0.4 m median accuracy; hence SpotFi achieves the same accu-

racy as these prior systems while using commodity WiFi cards on both the AP and the client and nothing else. To the best of our knowledge, no other localization system that works only with information that is already exposed by commodity WiFi cards and with no war-driving can achieve even sub-meter accuracy.

4.3.2 High NLoS Deployments

Next, we evaluate SpotFi under stressful conditions. First we test SpotFi under conditions where only two or less number of APs have a decent direct path from the client due to blockages. To do this, we use 23 locations in our testbed where this condition holds based on our ground truth estimates. Much prior work [1, 8, 2] does not study such stressful scenarios, so it is hard to comment on how they might perform. However we do compare our system against our ArrayTrack implementation. We repeat the same localization experiments as before and plot the CDF of localization errors in Fig. 7(b).

As expected, localization error is worse in these scenarios. SpotFi achieves a median accuracy of 1.6 m whereas ArrayTrack degrades to 3.5 m. The reason is that the direct path AoA estimation error is higher in such scenarios relative to the indoor office deployment scenario. But SpotFi's super-resolution algorithm still works much better compared to the AoA estimation algorithm which ArrayTrack uses, and the accuracy is still acceptable for many applications. Also, SpotFi's unique direct path likelihood estimates evaluated at each AP result in less weight for measurements from APs which do not have strong direct path to the target thus reducing the inaccuracy in location estimation.

4.3.3 Corridors

Another feature of indoor spaces such as offices and malls are corridors, and these are precisely the areas where people desire localization capabilities for indoor navigation. Corridors are challenging, especially narrow ones, because it is unlikely that a particular target location is surrounded by APs such that the target's AoA can be measured from diverse vantage points. Typically APs are deployed just along the side wall of the corridor and many of the AoA measure-

ments at each AP will be very close to each other due to geometry of the corridors. Also, even if the client has more APs in LoS, they are generally farther away than when compared to indoor office deployments. This has an important implication on localization accuracy since it leads to scenarios where many APs have inaccurate and correlated AoA measurements and it becomes harder for some AP to correct the errors of other APs.

We evaluate SpotFi under such a scenario in our testbed. Specifically we look at the target locations in the two corridors that we see in Fig. 6. There are 25 points overall in such locations. We repeat the localization experiments as described above and compute the localization error for both SpotFi and ArrayTrack. Again as seen from Fig. 7(c), the median localization error for SpotFi is around a meter, whereas ArrayTrack's error worsens to 4 m.

Improved localization accuracy of SpotFi is due to two factors. First, SpotFi resolves multipath more accurately with the same number of antennas compared to schemes which use only the relative phase information between the antennas. Second, SpotFi's novel localization algorithm which accurately identifies the direct path between the target and the APs among the estimated multipath components. We now test individually the significance of these factors next.

4.4 Deep Dive into SpotFi

4.4.1 AoA Estimation Accuracy

Method: The goal here is to show that the super-resolution algorithm of SpotFi provides a much more accurate set of AoA estimates for all propagation paths. However, we have ground truth AoA for only the direct path between the target and the AP. So, we measure the accuracy of the AoA estimation algorithm by measuring the difference between ground truth direct path AoA and estimated AoA that is closest to this ground truth. This way, we remove the effect of direct path selection process which identifies the direct path AoA, which we will quantify in the next section separately.

For this evaluation, we classify the data into Line of Sight (LoS) scenarios and Non-Line of Sight (NLoS) scenarios. For AoA estimation error purposes, we define NLoS scenario for an AP as a scenario where there is a strong blocking object like a wall obstructing the line joining the target and the AP and hence the direct path is significantly weaker or non-existent compared to reflected paths. All other scenarios are declared as LoS. We refer to the AoA estimation algorithm used in localization systems like Phaser [8], and described in Sec. 3.1.1, as MUSIC-AoA algorithm.

Analysis: Fig. 8(a) plots the CDFs for AoA estimation error for all the APs. In LoS cases, SpotFi achieves median AoA accuracy of 2.4 degrees better than that achieved by MUSIC-AoA. In NLoS scenarios the accuracy is even better, achieving an improvement of nearly 5.2 degrees in direct path AoA error. The reason is that in LoS scenarios even MUSIC-AoA works reasonably well since the direct path is likely the strongest of all paths. But in NLoS scenarios many reflected paths will have the same or higher power
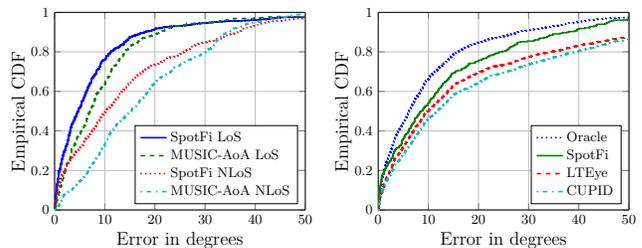
(a) AoA Estimation error   (b) Direct path AoA selection error

Figure 8: Reconstructing the effect of different components of SpotFi in improving the localization accuracy: (a) plots the CDFs of SpotFi's AoA estimation error with respect to the MUSIC-AoA scheme which only models the phase shifts across antennas to estimate AoA. The comparison is made individually for cases where there is a strong obstruction like wall between the target and AP (NLoS) and in cases where there is none (LoS). (b) plots the CDFs of SpotFi's error in selecting the AoA corresponding to direct path AoA and compares with schemes used in CUPID [23] and LTEye [6] systems.

compared to the direct path, and standard AoA-based MUSIC with three antennas does not have sufficient resolution to estimate the direct path's AoA accurately.

4.4.2 Direct Path AoA Selection Accuracy

Method: Next, we compute the efficacy of SpotFi's likelihood based technique to pick out the direct path from a bunch of candidate paths as described in section 3.2. To demonstrate the merit of SpotFi's direct path selection algorithm, we compare it with three other AoA selection algorithms used in prior work [6, 23]:

- LTEye [6] declares the direct path as the one with smallest ToF. However note that, unlike LTEye, we do not have access to the actual ToF between the target and the AP as they are not time synchronized. However, path with smallest estimated ToF also corresponds to the path with smallest actual ToF because lack of synchronization just adds a constant delay to ToFs of all the paths.
- CUPID [23] declares the AoA with the largest value in the MUSIC spectrum as the the direct path AoA.
- Oracle selection algorithm which chooses the AoA closest to the ground truth direct path AoA.

Analysis: We define AoA selection error as the difference between the ground truth direct path AoA and AoA selected by the direct path selection algorithm described in Sec. 3.2. Note that all of these schemes are working with the AoA estimates from SpotFi's super-resolution algorithm, not the standard AoA-based MUSIC. Fig. 8(b) plots the CDF of the error for all the four schemes considering all the deployment scenarios. SpotFi's technique is the best scheme relative to the Oracle, whereas the smallest delay based AoA selection has 80th percentile error 10 degrees worse than SpotFi. Picking the AoA with largest value in MUSIC spectrum, or equivalently direction with largest energy, performs the worst. This is because in a typical indoor environment with objects and humans blocking direct path to the AP, the direct signal can get heavily attenuated compared to signals traveling along reflected paths.
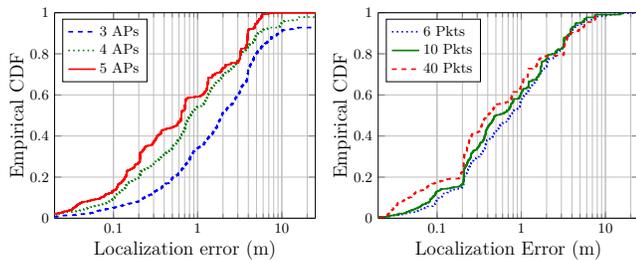
(a) localization error with different AP densities

(b) localization error with different number of packets used for localization

Figure 9: (a) plots CDFs of SpotFi's localization error as we vary the number of APs that can hear the target. This represents deployment scenarios with different densities of WiFi APs. (b) plots the CDFs of SpotFi's localization error as we vary the number of packets used for localizing the target.

4.4.3 Impact of WiFi Deployment Densities

Method: We emulate different WiFi AP densities by localizing using CSI from only random subsets of the APs. We vary the number of APs that can hear the target between three and five to demonstrate the performance of SpotFi with increasing deployment density.

Analysis: Fig. 9(a) plots the localization error for target locations in indoor office deployment. We observe that accuracy improves with increasing number of APs as expected. As we have more APs, it becomes easier to find at least a few APs that have decent direct path signal from the target and thus eliminate spurious estimates. We observe that SpotFi achieves median localization error of 0.6 m, 0.8 m, and 1.9 m with five, four and three APs respectively. So, even with four APs, SpotFi outperforms practical implementation of ArrayTrack with six APs. We observed a big improvement when we increased the number of APs from three to four, and much smaller improvement as we added more APs. The reason for diminishing improvements in accuracy with increasing number of APs is that once we have two strong direct path measurements, additional direct path measurements, if accurate, will only help in improving location accuracy incrementally, and if inaccurate, will effectively not be considered due to SpotFi's robust localization algorithm.

4.4.4 Impact of Number of Packets

Method: Using multiple packets improves SpotFi's accuracy in identifying the direct path among the estimated paths as described in section 3.2. Ideally, we want a localization system to perform well using small number of packets so that localization delay is minimized and the system works robustly even when the target can only send a small amount of traffic. We vary the number of packets SpotFi uses for localization from 6 packets to 40 packets and plot the effect on the localization accuracy for the target locations in indoor office deployment in Fig. 9(b).

Analysis: We observe from Fig. 9(b) that even with 10 packets, SpotFi accurately identifies the direct path and achieves a median localization accuracy of 0.5 m compared to 0.4 m obtained using 40 packets. So, SpotFi works well with just

10 CSI measurements; in other words, SpotFi requires the target to transmit just 10 packets for accurate localization.

5. CONCLUSION

SpotFi provides accurate indoor localization services using commercial off-the-shelf WiFi NICs with three antennas and with an accuracy comparable to state-of-the-art localization systems with large antenna arrays which are not suitable for wide deployments. SpotFi's techniques for AoA estimation and direct path filtering however are more widely applicable for other problems such as device free localization, gesture recognition and motion tracing. Exploring them is future work.

6. REFERENCES

[1] J. Xiong and K. Jamieson, "Arraytrack: A fine-grained indoor location system," NSDI '13.

[2] S. Kumar, S. Gil, D. Katabi, and D. Rus, "Accurate indoor localization with zero start-up cost," MobiCom '14.

[3] P. Bahl and V. N. Padmanabhan, "Radar: An in-building rf-based user location and tracking system," INFOCOM 2000.

[4] K. Chintalapudi, A. Padmanabha Iyer, and V. N. Padmanabhan, "Indoor localization without the pain," MobiSys '05.

[5] B. Ferris, D. Fox, and N. Lawrence, "Wifi-slam using gaussian process latent variable models," IJCAI '07.

[6] S. Kumar, E. Hamed, D. Katabi, and L. Erran Li, "Lte radio analytics made easy and accessible," SIGCOMM '14.

[7] R. O. Schmidt, "Multiple emitter location and signal parameter estimation," *IEEE Trans. on Antennas and Propagation 1986*.

[8] J. Gjengset, J. Xiong, G. McPhillips, and K. Jamieson, "Phaser: Enabling phased array signal processing on commodity wifi access points," MobiCom '14.

[9] A. Paulraj, V. Reddy, T. Shan, and T. Kailath, "Performance analysis of the music algorithm with spatial smoothing in the presence of coherent sources," IEEE MILCOM 1986.

[10] K. Wu, J. Xiao, Y. Yi, M. Gao, and L. Ni, "Fila: Fine-grained indoor localization," INFOCOM '12.

[11] H. Lim, L.-C. Kung, J. C. Hou, and H. Luo, "Zero-configuration, robust indoor localization: Theory and experimentation," 2005.

[12] A. Goswami, L. E. Ortiz, and S. R. Das, "Wigem: A learning-based approach for indoor localization," CoNEXT '11.

[13] P. Bahl, V. N. Padmanabhan, and A. Balachandran, "Enhancements to the radar user location and tracking system," tech. rep., 2000.

[14] M. Youssef and A. Agrawala, "The horus wlan location determination system," MobiSys '05.

[15] S. Sen, B. Radunovic, R. R. Choudhury, and T. Minka, "You are facing the mona lisa: Spot localization using phy layer information," MobiSys '12.

[16] R. Nandakumar, K. K. Chintalapudi, and V. N. Padmanabhan, "Centaur: locating devices in an office environment," MobiCom '12.

[17] M. Azizyan, I. Constandache, and R. Roy Choudhury, "Surroundsense: mobile phone localization via ambience fingerprinting," in *MobiCom '09*.

[18] A. Rai, K. K. Chintalapudi, V. N. Padmanabhan, and R. Sen, "Zee: Zero-effort crowdsourcing for indoor localization," Mobicom '12.

[19] Z. Yang, C. Wu, and Y. Liu, "Locating in fingerprint space: Wireless indoor localization with little human intervention," Mobicom '12.

[20] H. Liu, Y. Gan, J. Yang, S. Sidhom, Y. Wang, Y. Chen, and F. Ye, "Push the limit of wifi based localization for smartphones," MobiCom '12.

[21] H. Wang, S. Sen, A. Elgohary, M. Farid, M. Youssef, and R. R. Choudhury, "No need to war-drive: unsupervised indoor localization," MobiSys '12.

[22] M. Youssef and A. Agrawala, "Small-scale compensation for wlan location determination systems," in *IEEE Wireless Communications and Networking, 2003*.

[23] S. Sen, J. Lee, K.-H. Kim, and P. Congdon, "Avoiding multipath to revive inbuilding wifi localization," MobiSys '13.

[24] K. Joshi, S. Hong, and S. Katti, "Pinpoint: localizing interfering radios," NSDI '13.

[25] D. Niculescu and B. Nath, "Vor base stations for indoor 802.11 positioning," MobiCom '04.

[26] L. Atzori, A. Iera, and G. Morabito, "The internet of things: A survey," *Comput. Netw. '10*.

[27] P. Chen, P. Ahammad, C. Boyer, S. i Huang, L. Lin, E. Lobaton, M. Meingast, S. Oh, S. Wang, P. Yan, A. Y. Yang, C. Yeo, L. chung Chang, J. D. Tygar, and S. S. Sastry, "Citric: A low-bandwidth wireless camera network platform," ICDSC '08.

[28] *nest*. https://nest.com/.

[29] M. Youssef, A. Youssef, C. Rieger, U. Shankar, and A. Agrawala, "Pinpoint: An asynchronous time-based location determination system," MobiCom '06.

[30] S. A. Golden and S. S. Bateman, "Sensor measurements for wi-fi location with emphasis on time-of-arrival ranging," *IEEE Trans. on Mobile Computing*, 2007.

[31] A. T. Mariakakis, S. Sen, J. Lee, and K.-H. Kim, "Sail: Single access point-based indoor localization," MobiCom '14.

[32] A. Marcaletti, M. Rea, D. Giustiniano, V. Lenders, and A. Fakhreddine, "Filtering noisy 802.11 time-of-flight ranging measurements," CoNEXT '14.

[33] M. Ciurana, F. Barcelo-Arroyo, and F. Izquierdo, "A ranging system with ieee 802.11 data frames," in *IEEE Radio and Wireless Symposium, 2007*.

[34] S. Lanzisera, D. Zats, and K. S. Pister, "Radio frequency time-of-flight distance measurement for low-cost wireless sensor localization," *IEEE Sensors Journal*, 2011.

[35] J. Xiong, K. Jamieson, and K. Sundaresan, "Synchronicity: Pushing the envelope of fine-grained localization with distributed mimo," HotWireless '14.

[36] F. Zhao, W. Yao, C. C. Logothetis, and Y. Song, "Super-resolution toa estimation in ofdm systems for indoor environments," in *IEEE International Conference on Networking, Sensing and Control, 2007*.

[37] V. Amendolare, D. Cyganski, and R. J. Duckworth, "Transactional array reconciliation tomography for precision indoor location," *IEEE Trans. on Aerospace and Electronic Systems*, 2014.

[38] S. Venkatraman and J. Caffery, "Hybrid toa/aoa techniques for mobile location in non-line-of-sight environments," in *IEEE Wireless Communications and Networking Conference, 2004*.

[39] A. Cavanaugh, M. Lowe, D. Cyganski, and R. Duckworth, "Wpi precision personnel location system: Rapid deployment antenna system and sensor fusion for 3d precision location," in *Institute of Navigation-International Technical Meeting 2010*.

[40] H. Rahul, H. Hassanieh, and D. Katabi, "Sourcesync: a distributed wireless architecture for exploiting sender diversity," *ACM SIGCOMM CCR '11*.

[41] M. Wax and A. Leshem, "Joint estimation of time delays and directions of arrival of multiple reflections of a known signal," IEEE ICASSP 1996.

[42] A.-J. Van Der Veen, M. C. Vanderveen, and A. J. Paulraj, "Joint angle and delay estimation using shift-invariance properties," *IEEE Signal Processing Letters 1997*.

[43] M. C. Vanderveen, A.-J. Van der Veen, and A. Paulraj, "Estimation of multipath parameters in wireless communications," *IEEE Trans. on Signal Processing 1998*.

[44] M. Vanderveen, B. Ng, C. Papadias, and A. Paulraj, "Joint angle and delay estimation (jade) for signals in multipath environments," in *Conference Record of the Thirtieth Asilomar Conference on Signals, Systems and Computers, 1996*.

[45] Y.-Y. Wang, J.-T. Chen, and W.-H. Fang, "Tst-music for joint doa-delay estimation," *IEEE Trans. on Signal Processing*, 2001.

[46] J. Picheral and U. Spagnolini, "Shift invariance algorithms for the angle/delay estimation of multipath space-time channel," in *Vehicular Technology Conference, 2001*.

[47] D. Inserra and A. M. Tonello, "A frequency-domain los angle-of-arrival estimation approach in multipath channels," *IEEE Trans. on Vehicular Technology*, 2013.

[48] J.-T. Chen, J. Kim, and J.-W. Liang, "Multichannel mlse equalizer with parametric fir channel identification," *IEEE Trans. on Vehicular Technology*, 1999.

[49] G. G. Raleigh and T. Boros, "Joint space-time parameter estimation for wireless communication channels," *IEEE Trans. on Signal Processing*, 1998.

[50] J. He, M. Swamy, and M. O. Ahmad, "Joint space-time parameter estimation for underwater communication channels with velocity vector sensor arrays," *IEEE Trans. on Wireless Communications*, '12.

[51] H. Yamada, M. Ohmiya, Y. Ogawa, and K. Itoh, "Superresolution techniques for time-domain measurements with a network analyzer," *IEEE Trans. on Antennas and Propagation*, 1991.

[52] M. Ascione, A. Buonanno, M. D'Urso, L. Angrisani, and R. Schiano Lo Moriello, "A new measurement method based on music algorithm for through-the-wall detection of life signs," *IEEE Trans. on Instrumentation and Measurement*, 2013.

[53] J. Wang, F. Adib, R. Knepper, D. Katabi, and D. Rus, "Rf-compass: robot object manipulation using rfids," MobiCom '13.

[54] J. Wang and D. Katabi, "Dude, where's my card?: Rfid positioning that works with multipath and non-line of sight," in *ACM SIGCOMM CCR '13*.

[55] J. Wang, D. Vasisht, and D. Katabi, "Rf-idraw: Virtual touch screen in the air using rf signals," SIGCOMM '14.

[56] S. Gezici, Z. Tian, G. B. Giannakis, H. Kobayashi, A. F. Molisch, H. V. Poor, and Z. Sahinoglu, "Localization via ultra-wideband radios: a look at positioning aspects for future sensor networks," *IEEE Signal Processing Magazine*, 2005.

[57] A. Ward, A. Jones, and A. Hopper, "A new location technique for the active office," *IEEE Personal Communications*, 1997.

[58] R. K. Harle and A. Hopper, "Deploying and evaluating a location-aware system," MobiSys '05.

[59] N. B. Priyantha, A. Chakraborty, and H. Balakrishnan, "The cricket location-support system," MobiCom 2000.

[60] A. Harter, A. Hopper, P. Steggles, A. Ward, and P. Webster, "The anatomy of a context-aware application," *Wireless Networks*, 2002.

[61] A. Savvides, C.-C. Han, and M. B. Strivastava, "Dynamic fine-grained localization in ad-hoc networks of sensors," MobiCom '01.

[62] R. Want, A. Hopper, V. Falcao, and J. Gibbons, "The active badge location system,"

[63] A. Harter and A. Hopper, "A distributed location system for the active office," *IEEE Network*, 1994.

[64] E. Aitenbichler and M. Muhlhauser, "An ir local positioning system for smart items and devices," in *Distributed Computing Systems Workshops, 2003*.

[65] P. Hu, L. Li, C. Peng, G. Shen, and F. Zhao, "Pharos: Enable physical analytics through visible light based indoor localization," HotNets '13.

[66] Y.-S. Kuo, P. Pannuto, K.-J. Hsiao, and P. Dutta, "Luxapose: Indoor positioning with mobile phones and visible light," MobiCom '14.

[67] *iBeacon*. https://developer.apple.com/ibeacon/.

[68] D. Halperin, W. Hu, A. Sheth, and D. Wetherall, "Tool release: Gathering 802.11n traces with channel state information," *ACM SIGCOMM CCR '11*.

[69] N. Czink, M. Herdin, H. Ozcelik, and E. Bonek, "Number of multipath clusters in indoor mimo propagation environments," *Electronics Letters '04*.

[70] V. Erceg, L. Schumacher, P. Kyritsi, and et al, "Tgn channel models," *Tech. Rep. IEEE P802.11, Wireless LANs, 2004*.

[71] A. Goldsmith, *Wireless communications*. Cambridge university press, 2005.

[72] S. Boyd and L. Vandenberghe, *Convex Optimization*. New York, NY, USA: Cambridge University Press, 2004.