ToneTrack: Leveraging Frequency-Agile Radios for Time-Based Indoor Wireless Localization

Jie Xiong
University College London
j.xiong@cs.ucl.ac.uk

Karthikeyan Sundaresan
NEC Laboratories America
karthiks@nec-labs.com

Kyle Jamieson
University College London
k.jamieson@cs.ucl.ac.uk

Abstract

Indoor localization of mobile devices and tags has received much attention recently, with encouraging fine-grained localization results available with enough line-of-sight coverage and hardware infrastructure. Some of the most promising techniques analyze the time-of-arrival of incoming signals, but the limited bandwidth available to most wireless transmissions fundamentally constrains their resolution. Frequency-agile wireless networks utilize bandwidths of varying sizes and locations in a wireless band to efficiently share the wireless medium between users. ToneTrack is an indoor location system that achieves sub-meter accuracy with minimal hardware and antennas, by leveraging frequency-agile wireless networks to increase the effective bandwidth. Our novel signal combination algorithm combines time-of-arrival data from different transmissions as a mobile device hops across different channels, approaching time resolutions previously not possible with a single narrowband channel. ToneTrack's novel channel combination and spectrum identification algorithms together with the triangle inequality scheme yield superior results even in non-line-of-sight scenarios with one to two walls separating client and APs and also in the case where the direct path from mobile client to an AP is completely blocked. We implement ToneTrack on the WARP hardware radio platform and use six of them served as APs to localize Wi-Fi clients in an indoor testbed over one floor of an office building. Experimental results show that ToneTrack can achieve a median 90 cm accuracy when 20 MHz bandwidth APs overhear three packets from adjacent channels.

Categories and Subject Descriptors

C.2.1 [**Computer-Communication Networks**]: Network Architecture and Design—*Wireless communication*

Keywords

Indoor Wireless Location; Time (Difference) of Arrival (ToA/TDoA); Channel Combination; CSI Combination; Bandwidth Increment; MUSIC Spectrum Identification; Time Synchronization; Distributed MIMO; Triangle Inequality

MobiCom'15, September 7–11, 2015, Paris, France.
ACM 978-1-4503-3619-2/15/09.
http://dx.doi.org/10.1145/2789168.2790125.

Physical layer	Bandwidth	Raw resolution
802.11a/g Wi-Fi	20 MHz	15 m
802.11n Wi-Fi	40	7.5
802.11ac Wi-Fi	$<$ 160	$>$ 1.9
Ultra-wideband	$>$ 500	$<$ 60 cm

Table 1: Popular physical layers used in localization, their frequency bandwidth, and the raw sample spatial resolution each offers—the distance light travels between sampling instants at that bandwidth: Raw resolution = Speed of light / Bandwidth.

1. INTRODUCTION

Recently, indoor wireless localization systems have broken the meter accuracy barrier both for Wi-Fi devices [18, 58, 59] and RFID tags [52, 53, 62], but to achieve these results, require some combination of many access points (APs) and antennas, very long antenna arrays, and/or an RF environment without too many obstacles blocking client-AP lines of sight.

While recent systems have broken the meter accuracy barrier with angle-of-arrival (AoA) and other types of signal processing analysis, time-of-arrival (ToA) analysis promises to improve accuracy even further. ToA has a particular challenge, however, as shown in Table 1: for a typical 802.11a/g Wi-Fi channel with only 20 MHz bandwidth, the signal is sampled once every 50 nanoseconds, during which the signal travels a full 15 meters. As the next rows of the table show, later 802.11n/ac standards enhance this resolution, but still achieve just 1.9 meters of raw sample resolution. Super-resolution spectral signal processing algorithms such as MUSIC [25, 41] and matrix-pencil [39] can enhance this raw sample resolution by an approximate factor of $2\times$, but still achieve an accuracy proportional to the raw sample spatial resolution shown in Table 1, limiting the utility of ToA analysis. Even ultra-wideband (UWB) systems that sample at a rate of 500 MHz and up achieve just 60 cm raw spatial resolution. Focusing on ToA analysis, this paper questions whether we can do better.

The opportunity we leverage in this work is that tomorrow's wireless networks will make adaptive and opportunistic use of a large variety of frequency bandwidths, ranging from narrow 5 MHz channels intended for the exclusive use of one mobile user at a time, to expansive 160 MHz channels shared between users with CSMA. Indeed, the use of narrow frequency-bandwidth channels is now commonplace: Wi-Fi [6, 9, 46] and cellular systems divide the wireless medium into fine-grained time-frequency blocks, conferring many benefits such as reducing fixed-airtime MAC overheads, increasing signal-to-noise ratio (SNR), and allowing for channel assignment algorithms to optimize throughput for many users. Furthermore, the use of wide-bandwidth channels has also emerged.

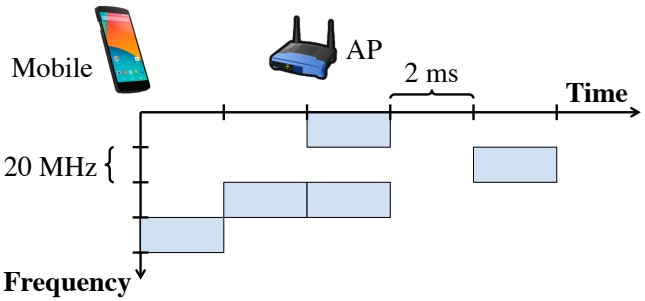


Figure 1: A Wi-Fi mobile hops across 80 MHz of bandwidth in 10 ms in order to avoid other competing Wi-Fi users and in-band interference. Cellular LTE mobiles use a similar strategy for similar reasons. ToneTrack leverages in-band frequency hopping to improve indoor localization accuracy.

The 802.11ac standard [16] specifies transmission bandwidths from 20 to 160 MHz, even allowing two non-contiguous 80 MHz channels to be aggregated together as one 160 MHz channel. Dynamic Frequency Selection (DFS) in the 5.250–5.725 MHz band lets Wi-Fi radios hop channels to avoid any nearby military radar.

In this paper, we present *ToneTrack*, an indoor localization system that leverages frequency-agile wireless networks to enhance the accuracy of indoor localization. ToneTrack measures the ToA of a client's transmission at pairs of APs in the network. In order to do this, it analyzes the correlation between incoming signals on different subcarriers as MUSIC does, but in the frequency domain. This allows ToneTrack to achieve higher time-of-arrival accuracy than simply looking at the sample index of packet detection or channel impulse response. But as noted above, even with super-resolution MUSIC scheme, frequency bandwidth still limits the resolution that ToA algorithms can achieve.

To increase the bandwidth available for time-based localization, ToneTrack contributes a novel signal combination scheme that combines data from a device as it hops across different channels in a frequency band, as shown in Figure 1.[1] The result is that ToneTrack can achieve gains in time resolution that are proportional to the number of channels hopped across when transmitting within a channel coherence time.

After extracting a ToA profile of the mobile device's signal from each AP, ToneTrack analyzes each profile individually. Even when multipath reflections arrive too close in time to the direct path and super-resolution schemes reach their resolution limits, failing to resolve all the paths correctly, ToneTrack is still able to identify the useful data therein, retrieving relatively accurate information despite inaccuracy in the overall ToA profile. Here novel peak classification algorithms identify the accurate direct-path peak in the time-of-arrival profile and retain it for further processing.

Lastly, ToneTrack compares TDoA readings across pairs of APs in the network in order to estimate and refine the mobile client's location. Most prior indoor localization work cope with multipath reflections when both reflection paths and direct path exist. The direct path signal may get attenuated but does exist. However, the direct path signal sometimes gets 100% blocked which is even more challenging. ToneTrack employs the classical triangle inequality property to identify the APs with whose direct path is completely blocked, improving accuracy in this most challenging situation. Then clustering, outlier rejection, and averaging complete the processing chain, yielding the location estimate from a mobile client's transmission. ToneTrack does not require any offline training: preamble data from one to three packets suffice, making the approach amenable to real-time tracking.

[1]Hopping between channels within a frequency band allows deployment with just a single kind of antenna.

Contributions. To summarize, ToneTrack contributes the following novel design elements:

1. A frequency (tone) combining algorithm that allows a ToA/TDoA method to increase the bandwidth it may utilize for finer accuracy without increasing the radio's sampling rate (§2.3).
2. Retrieve useful information from the inaccurate ToA spectrum profile even when the super-resolution scheme reaches the resolution limit (§2.4).
3. A triangle inequality-based method together with outlier rejection scheme for identifying and discarding the AP to which a client's line of sight transmission is completely blocked (§2.5).

Roadmap. The rest of this paper begins with our system design (§2) and implementation (§3). Our evaluation (§4) in an indoor 20×25 meter office testbed demonstrates a 90 cm median localization accuracy with four APs, each equipped with one antenna and overhearing three packets transmitted at adjacent channels with 20 MHz bandwidth. We survey related work in the area in Section 5 before concluding (§6).

2. DESIGN

This section presents the design of ToneTrack, starting with a system description (§2.1) before delving into ToneTrack's constituent parts: super-resolution ToA processing (§2.2), channel combination (§2.3), spectrum identification (§2.4), and multi-AP data fusion (§2.5).

2.1 System design

ToneTrack is designed as a passive system that listens to mobile clients' transmissions at nearby APs. Thus the system requires no additional wireless channel overhead for deployment in a production wireless local-area network. Figure 2 shows the high-level system design: upon hearing multiple packet transmissions on different channels from a mobile device, an AP forwards the packets to the backend server over a backhaul wired network, appended with timestamps. Then, once the backend server receives this data within a channel coherence time, it passes them to the channel combination step described in Section 2.3 to generate a high-resolution time of arrival profile. Next, novel algorithms determine whether the resulting ToA profile is in fact accurate, or alternately, contains an accurate part useful for localization, even when the overall ToA profile is inaccurate (§2.4). After that, the ToneTrack controller combines the ToA information collected at pairs of APs into TDoA estimates. In Figure 2, the hyperbolic curve labeled "AP 1/2" denotes the possible loci of the mobile based on AP 1 and AP 2's TDoA and the hyperbolic curve labeled "AP 1/3" denotes the possible loci of the mobile based on AP 1 and AP 3's TDoA. Finally, the server processes the TDoA estimates across pairs of APs using geometrical reasoning (triangle inequality), clustering and outlier rejection schemes (§2.5), yielding a final location estimate.

2.2 ToA estimation

Once a client's transmission arrives at an AP, ToneTrack measures the time of arrival (*ToA*) of a client's transmission at one AP: this section describes this process in detail.

2.2.1 Primer: MUSIC in the frequency domain

We begin with the classical MUSIC algorithm [25, 41], which models the multipath indoor radio propagation channel $h(t)$ as the sum

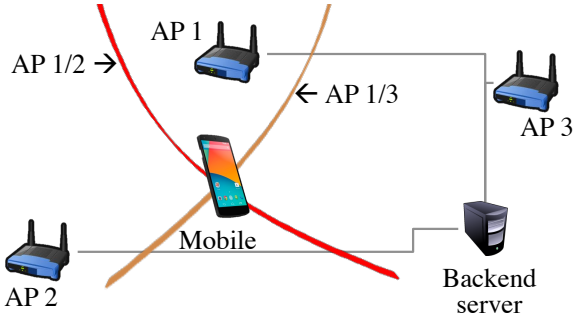


Figure 2: *High-level design of ToneTrack.* APs overhear a packet transmission from a mobile and pass the packets to the backend server to run a time-of-arrival (ToA) estimation algorithm, combining the resulting hyperbolic loci (labeled in the figure with their originating AP pairs) for a location estimate.

of D attenuated and delayed impulse responses:

$$h(t) = \sum_{k=1}^{D} \alpha_k \delta(t - \tau_k). \tag{1}$$

Here α_k and τ_k are the complex attenuation and propagation delay of the kth path. For simplicity, in this section we describe ToneTrack's operation over one Wi-Fi channel. Later sections generalize to multiple Wi-Fi channels.

Processing starts with the per-subcarrier channel response of Equation 1 in the frequency domain:

$$\mathbf{H}[f_n] = \sum_{k=1}^{D} \alpha_k e^{-j2\pi(f_0 + n\Delta f)\tau_k}. \tag{2}$$

Here, f_n and Δf are the carrier frequency and the size of subcarrier bandwidth, respectively. We estimate $\mathbf{H}[f_n]$ by taking the DFT of the received 64-sample 802.11 long training symbol and dividing, per-subcarrier, by the known transmitted long training symbol. We denote this estimate as $\hat{\mathbf{H}}[f_n]$. In 802.11a/g, 52 out of 64 subcarriers contain preamble information; we employ all of them in the processing that follows. The *subcarrier correlation matrix* $\mathbf{R_{HH}}$ then measures phase changes between different subcarriers:

$$\mathbf{R_{HH}} = E\{\hat{\mathbf{H}}[f_n]\hat{\mathbf{H}}^*[f_n]\}, \tag{3}$$

where the expectation is calculated across multiple OFDM symbols (spaced in time).

Suppose D copies (direct path and reelection paths) of a transmission $s_1, \ldots, s_D$ arrive at the AP's antenna at D respective times $t_1, \ldots, t_D$, and further suppose the OFDM symbol of the transmission contains M subcarriers ($M > D$) so all copies of the transmission can be captured. Eigenanalysis of the subcarrier correlation matrix $\mathbf{R_{HH}}$ at the AP then results in M eigenvalues associated respectively with M eigenvectors $\mathbf{E} = [\mathbf{e}_1\ \mathbf{e}_2\ \cdots\ \mathbf{e}_M]$. If we sort the eigenvalues in non-decreasing order, the smallest $M - D$ eigenvalues tend to correspond to background noise while the next D eigenvalues tend to correspond to the D incoming copies of the mobile's transmission. Based on this process, the corresponding eigenvectors in $\mathbf{E}$ are classified as noise subspace and signal subspace:

$$\mathbf{E} = \left[\overbrace{e_1\ \ldots\ e_{M-D}}^{\mathbf{E}_N}\ \overbrace{e_{M-D+1}\ \ldots\ e_M}^{\mathbf{E}_S}\right] \tag{4}$$

We refer to $\mathbf{E}_N$ as the *noise subspace* and $\mathbf{E}_S$ as the *signal subspace*. We define a *time steering vector* $\mathbf{a}(\tau)$ that represents the channel's response to a signal arriving at time τ:

$$\mathbf{a}(\tau) = \begin{bmatrix} 1 \\ \exp(-j2\pi\tau\Delta f) \\ \vdots \\ \exp(-j2(M-1)\pi\tau\Delta f) \end{bmatrix} \tag{5}$$

The *time steering vector* $\mathbf{a}(\tau)$ is in the signal subspace and is orthogonal to the noise subspace when τ exactly coincides with each time of arrival of the signal. The *MUSIC ToA spectrum* then measures the distance (in a vector space defined by the array correlation matrix above) between the time steering vector and the noise subspace, as τ varies, thus estimating the time arrival of multiple signals with a granularity of our own choosing:

$$P(\tau) = \frac{1}{\mathbf{a}(\tau)^H \mathbf{E}_N \mathbf{E}_N^H \mathbf{a}(\tau)}. \tag{6}$$

With the steering vector and noise subspace vector in the denominator, $P(\tau)$ generates peaks when the steering vector is orthogonal to the noise subspace vector which happens when τ coincides with the time of arrivals of the incoming signals.

Limitations of MUSIC's super-resolution capability. MUSIC is informally known as a *super-resolution* algorithm. The τ variable in Equation 6 can vary in steps of our own choosing smaller than the sampling period shown in Table 1. But this does not imply MUSIC is able to resolve multipaths with arbitrarily small time delay differences. The frequency bandwidth of the received transmission and background noise imposes a resolution limit independent of τ's step size chosen.

To probe this limit in a controlled experimental setting, we use the simple channel emulation setup shown in Figure 3. An RF splitter-combiner first splits a wired signal into two equal components, one of which travels over a longer cabled path than the other. A second RF splitter-combiner then combines the two signals together, where they are received and processed with MUSIC algorithm. We use different cable lengths[2] to control the relative path lengths, and attenuators to control the respective path signal strengths to the same level.

Decreasing the path length difference from 13.5 m (44 ft.) gradually to 2.7 m (8.8 ft.) results in the MUSIC pseudospectra shown in Figure 4. We see from the figure that MUSIC is able to resolve both paths quite accurately when their lengths are sufficiently different, but once the path length difference between the two signals is decreased to around six meters (20 ft.), MUSIC is not able to generate accurate pseudospectra anymore: its two spectrum peaks half-merge in Figure 4 (c) and (d), moving away from ground-truth. When we further decrease the path length difference, the two peaks fully merge into one peak in Figure 4 (e).

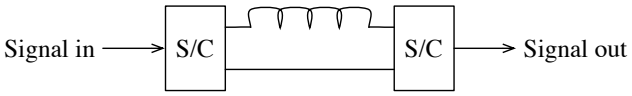


Figure 3: A simple two-tap channel emulator: An RF splitter-combiner ("S/C") splits an incoming signals into two branches: one travels over the longer (upper) cabled path, the other travels over the shorter (lower) cabled path. This network models an idealized wireless channel with two paths (one direct path and one reflection path), of varying differential path length.

[2]Because of lower transmission speed in cable, we translate cable length to equivalent air propagation distance. (The delay of a 1.8 m RG-58 cable is equivalent to 2.7 m propagation delay in the air).

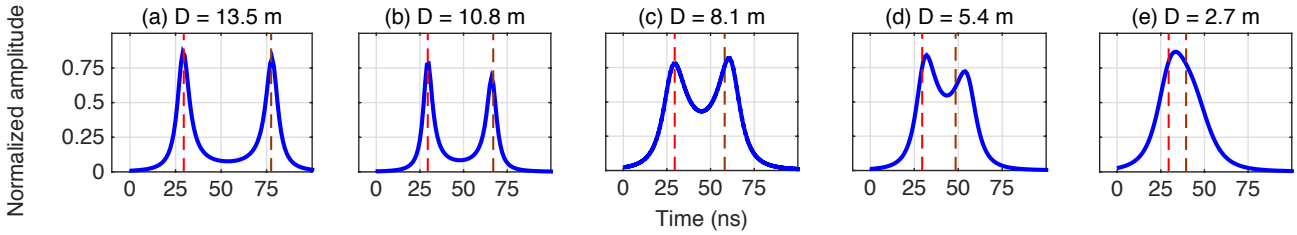


Figure 4: *MUSIC's resolution limit.* At 20 MHz bandwidth, MUSIC loses the ability to resolve two paths with a length difference of less than about six meters (20 ft). We denote the two ground-truth path lengths with dotted vertical lines.

2.3 Channel combination

To overcome the limitations of MUSIC's super-resolution capability noted above in Section 2.2.1, ToneTrack leverages the frequency agility of upcoming Wi-Fi, LTE, and white-space radios as they hop between different frequencies in short periods of time. Note that if frequency hopping happens within a channel coherence time, the ToA spectra generated are very similar, but each is a low-resolution picture of the ToA. The basic idea of ToneTrack's channel combination technique is to combine multiple frequency-agile transmissions from the client to form a virtual wider bandwidth transmission, without increasing the sampling rate. Since the effective *array aperture* of MUSIC's ToA estimate is proportional to the number of subcarriers measured (*i.e.*, the bandwidth), time resolution ought to scale linearly with bandwidth. However, naïvely concatenating data from two channels does not work: we need to align them in both time and frequency domain in order for the combined data to yield a better resolution in the ToA spectrum plot.

Alignment in time domain. While standard packet detection algorithms [40] can synchronize to sample level at typical baseband sampling rates, ToneTrack requires sub-sample level time alignment of the two overheard signals for combination.

Since the data are recorded at the same radio at different times, there are different fractional (sub-sample) time delays introduced to each set of data. In order to combine data, we need to remove this random time difference. As these two groups of data are recorded within a small time interval, the relative amplitudes of the peaks on the spectra are stable. We apply standard fractional interpolation methods [22] to align the two signals based on their respective ToA spectra. Sub-sample interpolation of the raw data in the time domain causes the whole ToA spectrum to move in time. One sample shift is corresponding to a spectrum movement of 50 ns at 20 MHz. We measure the time difference (in ns) of the largest peak position on each of the two spectra. We then align the two sets of data in the time domain, matching the two largest peaks to the same position by a corresponding sub-sample interpolation of the raw data. As demonstrated in Figure 5 (b1) and (b2), with a single signal, time domain alignment equalizes the slopes of the two groups of sub-carrier phases.

To see why this is the case, consider a measurement of phase in the frequency domain. Looking across subcarriers of separation Δ_f, the time-shifting property of the DFT

$$H[k]e^{2\pi j\tau_0 k/N} \overset{\mathscr{F}}{\longleftrightarrow} h[(n-\tau_0)_N] \tag{7}$$

tells us that if there is only one signal, phase at the AP changes linearly across subcarriers as $2\pi\Delta_f\tau_0/N$ where the slope of the phase is proportional to propagation time τ_0.

Alignment in frequency domain. Unfortunately, concatenating even time-aligned data from adjacent channels fails again, yielding completely inaccurate and noisy ToA spectra. We need to estimate

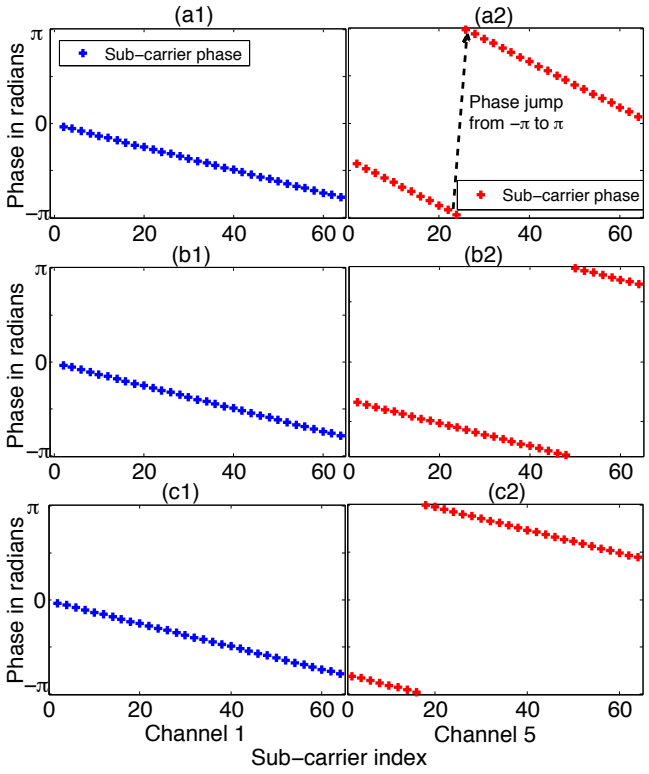


Figure 5: *ToneTrack's channel combination scheme.* Time domain alignment equalizes the slope of the phase in the frequency domain between channels, as shown in *(b1)* and *(b2)*. Subsequent frequency domain alignment removes the phase offset and enables successful concatenation of data as shown in *(c1)* and *(c2)*.

the phase of the sub-carrier just after the last sub-carrier of the first channel. Then we align the phase of the first sub-carrier of the second channel to the estimated one by subtracting the phase offset. This concept is demonstrated in Figure 5 (c1) and (c2) with data from two channels fully aligned in both time and frequency domains. With this step, the two groups of channel response data can now be concatenated to yield better resolution than any one alone. We form a larger virtual bandwidth without increasing the radio's sampling rate. When multipaths are present, the phase change becomes highly non-linear since it is the superposition of many paths of varying magnitude and phase. Our insight is that since this superposition remains continuous across channels in phase, ToneTrack can still align the two groups of data by matching the phase of the last sub-carriers in the first group of data to the first sub-carrier in the second group of data. With the scheme described here, we are

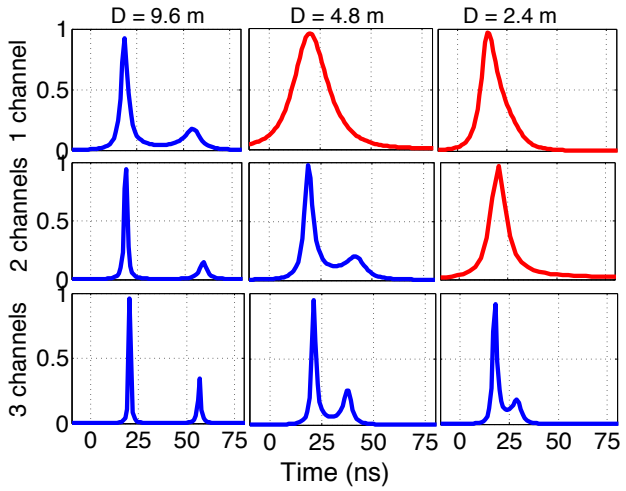


Figure 6: ToneTrack's channel combination scheme effectively increases the resolution capability of MUSIC, as tested by varying the path length difference d in our two-tap channel emulator. Red curves denote ToA spectra where peaks have problematically merged and MUSIC is not able to resolve them correctly.

able to concatenate multiple groups of data from adjacent channels seamlessly to perform like one single larger bandwidth channel.

2.3.1 *Channel combining microbenchmark*

We demonstrate the effectiveness of ToneTrack's channel combination with the microbenchmark results shown in Figure 6. In the first row, with one single 20 MHz channel, ToneTrack fails to resolve both signals when the path difference D between the signals decreases to 4.8 meters (15.8 feet). With the channel combination scheme applied with two channels, ToneTrack successfully resolves both two signals at $D = 4.8$ meters but fails to resolve when the difference decreases to 2.4 m. With three channels, ToneTrack is able to resolve two signals separated by only a 2.4 m (7.9 feet) path length difference. Our end-to-end localization results in Section 4.2 leverage this channel combining algorithm to markedly improve ToneTrack's accuracy level. The channel combination process at each AP is fully independent of the data fusion process later in Section 2.5 across multiple APs.

There is no limit on the number of channels can be employed for combination in ToneTrack. Our current implementation is based on 2.4 GHz and thus we employ channels 1, 5 and 9 for experiments. The spectrum range available in 2.4 GHz for Wi-Fi is small. We may include channel 11 to further add 10 MHz to the combination. The spectrum range in 5 GHz is much larger and more channels can be combined for higher accuracy.

2.3.2 *Overlapping and non-adjacent channels*

In the case of overlapping channels that may result when the mobile changes its center frequency by an amount less than the bandwidth of its transmissions, it is clear that ToneTrack's channel combining technique generalizes by averaging the channel information in the tones the two transmissions have in common. Then the two overlapping channels can be converted into two equivalent adjacent channels in terms of localization with the overlapping part removed from one channel.

We briefly discuss how our scheme can be generalized to sets of channels that are non-adjacent. The steering vector needs to be modified to reflect the different subcarrier separation between non-adjacent channels. This has the drawback of multiplying the number of peaks in the ToA spectrum, in a way analogous to the grating lobes problem RF-IDraw solves for AoA spectra [53]. Frequency domain alignment is challenging as it's not easy to estimate the correct amount of offset as the phase change is non-linear with strong multipaths. We leave the design and evaluation of non-adjacent channel combination in ToA as future work.

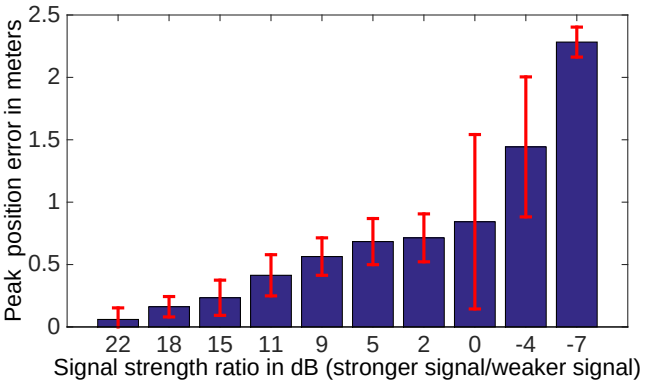


Figure 7: Peak position error when two peaks merged into one, as a function of the relative strength of the two peaks.

2.4 Spectrum identification

We now describe the processing ToneTrack performs on the ToA profile computed in §2.2 and §2.3 to determine whether the spectrum is accurate and if not, whether we can still retrieve relatively accurate direct-path information from the spectrum. We term this processing *spectrum identification*. As noted in Section 2.2, when the lengths of a line-of-sight path and a reflected path are too close to each other, MUSIC is unable to resolve the two signals correctly in the time domain on the pseudospectrum. This leads to either inaccurate pseudospectrum peak positions or multiple peaks merge. However, ToneTrack leverages the insight that we can sometimes still retrieve useful and relative accurate information from these *inaccurate* pseudospectra.

2.4.1 *Merged-signal peaks*

We first observe that when the two paths' peaks merge into one as shown in Figure 6, as long as the first (direct) path signal is stronger, the error in the peak position is still small. We use the simple two-tap channel emulator of Figure 3 to quantify this experimentally. In Figure 7, we vary the relative signal strength between the direct path and a reflection path 2.7 meters longer, starting from $+22$ dB (*i.e.*, direct path 22 dB stronger than reflection path) down to -7 dB (*i.e.*, reflection path 7 dB stronger than reflection path). The results in Figure 7 show that the error is well under one meter as long as the direct-path signal is stronger. The error increases significantly when the reflection path is stronger, up to 2.3 meters.

After we identify a merged peak, we measure the *skew direction* of the peak as shown in Figure 8 by finding the peak position and the two midpoints at which the peak amplitude falls by half (this is also known as the -3 *dB beamwidth*). By comparing the distance of the peak position to the two 3 dB beamwidth midpoints, we measure the direction of the peak's skew: a peak position falling to the right of the -3 dB beamwidth's midpoint as shown in Figure 8 (a) indicates that the first peak, which corresponds to the direct path, has merged into a later peak (which corresponds to a reflection path). ToneTrack identifies this merged peak as inaccurate and thus useless. The blue plot shows a spectrum skewing earlier in time (merged towards the direct-path peak). In this case

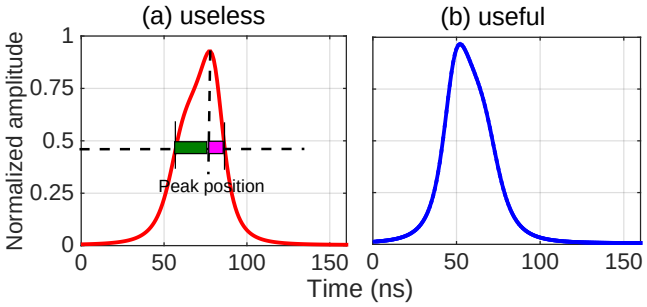


Figure 8: Merged-signal peaks. ToneTrack classifies useful spectra by the skew direction (earlier or later) of a merged peak: *(a)* the first (direct path) peak has merged into a later (reflection path) peak, or *(b)* a later (reflection path) peak has merged into the first (direct path) peak.

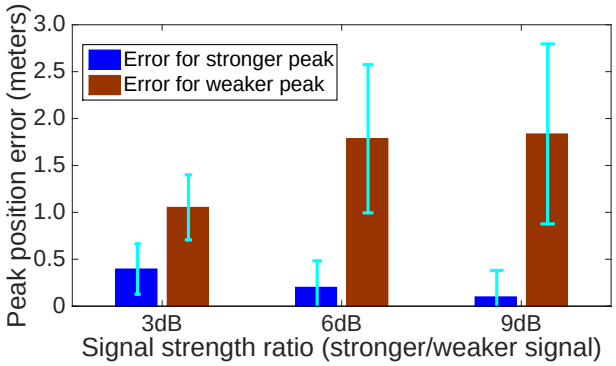


Figure 9: Peak error (translated from time to meters) for separated peaks in the simple two-tap channel emulator, when the direct path and the first-arriving reflection path are separated but arrive too close in time for MUSIC to accurately resolve.

the peak has a reasonably small error, and can thus still be kept for localization even it's a peak merged with two signals.

2.4.2 *Single-signal peaks*

If the two peaks are separated by more than the MUSIC resolution limit[3] as shown in Figure 10 (a1) and (a2), then MUSIC can accurately estimate their respective positions, and we feed the position of the first, direct-path peak to the next processing stage. But if the two peak positions are separated by less than the resolution limit as shown in Figure 10 (b1) and (b2), they fall into the zone that MUSIC is not able to resolve accurately.

Often, the direct path and reflection path signals have differing amplitudes. We anecdotally observe that even when the two peaks are too close for MUSIC to resolve, the larger peak on the pseudospectrum corresponding to the stronger signal is still quite accurate compared to the smaller peak. We validate this observation empirically in the simple two-tap channel emulator of Figure 3 with the following microbenchmark. We fix the path length difference between the direct-path and reflection-path signals to be 5.4 m (18 ft), then adjust the relative signal strength between the direct and reflection paths from 3 dB to 9 dB and show the peak position error in Figure 9. We see clearly that when the direct path signal is stronger, although MUSIC is not able to resolve both of them correctly, the error of the direct-path peak is quite small (less than 0.5 m). On the other hand, the smaller reflection path peak has a much larger

[3]We experimentally verified that at 20 MHz at medium-high SNRs, this resolution limit is stable and measured to be around 6 m.

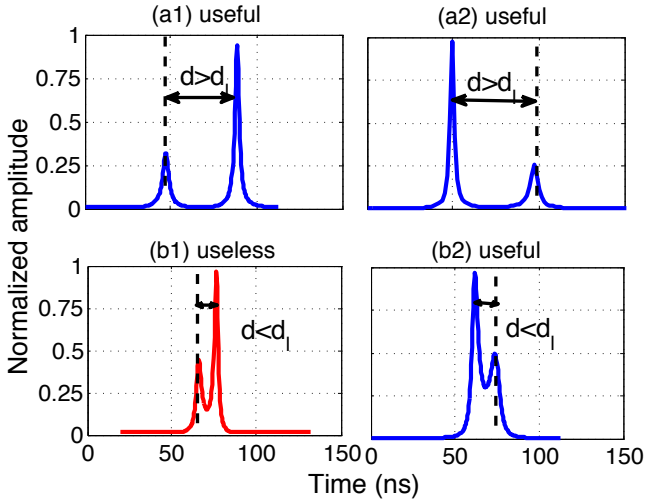


Figure 10: The peak separation d is greater than the resolution limit d_l, both (a1) and (a2) are kept. If the d is smaller than d_l, ToneTrack identifies useful ToA spectra by comparing their respective amplitudes and (b1) is discarded.

error, so we can still extract relatively accurate information from the MUSIC spectrum in these scenarios as we only care the direct-path peak. Referring to Figure 10 (b1) and (b2), when the separated peaks are closer than the resolution limit, ToneTrack compares the relative amplitudes of the two. If the amplitude of the first peak is greater than the second peak as in (b2), ToneTrack marks the ToA spectrum as useful; otherwise it discards the ToA spectrum such as in (b1). Referring again to the signals with a 3 dB difference in Figure 9, this bounds the error due to the presence of the second peak to well below one meter.

2.4.3 *Classifying peaks as merged or single-signal*

To apply the above spectrum identification technique, ToneTrack needs to estimate whether a certain peak arises from a single path or is the result of the respective peaks of multiple arrivals merging together in the ToA spectrum. Prior theoretical work [1] has shown that the beamwidth of the MUSIC spectrum is inversely proportional to the square root of SNR and the bandwidth of the signal. Consequently, within the SNR range ToneTrack operates at, a single-signal peak will be thinner compared to a merged peak, even if the merged peak originates from two closely-spaced signals. For example, the difference is apparent when we compare the red merged peaks in Figure 6 with blue peaks. ToneTrack thus measures the peak's -3 dB beamwidth $W_{-3\text{ dB}}$ and compares it with a threshold value W_t to make the decision:

$$W_{-3\text{ dB}} > W_t : \text{Merged peak.}$$
$$W_{-3\text{ dB}} \leq W_t : \text{Single peak.}$$

Using microbenchmarks measuring the impact of SNR and the path difference of the two signals on W_t, we experimentally determine the best value for W_t in Section 4.3, and show that it produces good end-to-end performance in our indoor testbed in Section 4.2.

2.4.4 *Algorithm (*Spectrum Identification*)*

As the preceding microbenchmarks show, useful and accurate information can still be retrieved even when MUSIC fails to resolve all the signals correctly, as long as information about the direct path peak is relatively accurate. In this section we summarize ToneTrack's *spectrum identification* algorithm, which comprises the pro-

cessing ToneTrack performs on each (possibly channel-combined, *cf.* §2.3) ToA spectrum from a single AP before passing that ToA spectrum on to the multi-AP data fusion step described next in Section 2.5.

Step 1. Isolate the first two peaks on the ToA spectrum as input to the algorithm. If the two peak positions are separated by greater than the resolution limit, then the first peak contains accurate direct-path distance information, so ToneTrack retains the spectrum and proceed to Step 3. Otherwise, the two peak positions are separated by a distance less than resolution limit (which MUSIC is not able to resolve accurately) so we proceed to Step 2:

Step 2. Compare the relative amplitudes of the two peaks. From the microbenchmarks, we know that as long as the direct path signal is stronger than the reflection path signal, the direct-path peak position will be more accurate. So ToneTrack retains the spectrum if and only if the first peak's amplitude exceeds the second's.

Step 3. Check whether the first peak is a single-signal peak or a merged peak (§2.4.3). ToneTrack retains the spectrum and the algorithm terminates in this step if the peak is a single-signal peak. Otherwise, we proceed to Step 4:

Step 4. Check the direction of the peak's skew (§2.4.1). ToneTrack retains the spectrum if and only if the peak is merged towards the direct path (left side).

After the above steps, only the useful peak remains. At this point ToneTrack sends the ToA spectrum to the multi-AP data fusion step described next.

2.5 Multi-AP data fusion

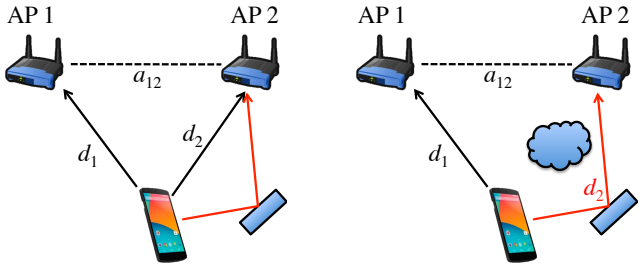
In this final stage of processing, ToneTrack converts measured ToAs from each AP into distance differences between pairs of APs, using these distance differences to estimate the mobile's location. Occasionally, the direct-path signal may be totally blocked, with only reflection signals detectable at the AP. We propose the following two methods to handle this very challenging scenario.

2.5.1 *Triangle inequality*

As shown in Figure 11(a), when both APs are able to resolve the direct-path signals from the mobile client, the distance estimates to AP 1 and AP 2 (d_1 and d_2, respectively), fit the following triangle inequality property:

$$d_1 + a_{12} \geq d_2, \tag{8}$$

where a_{12} is the distance between APs 1 and 2, which is known. However, when the direct path to AP 2 is completely blocked and only one or more reflection paths exist, as shown in Figure 11(b), the resulting distance estimates may violate this triangle inequality, *i.e.*, $d_1 + a_{12} < d_2$. Whenever we detect such a violation of the triangle inequality, we tag the violating AP (AP 2 in this example) as having its direct path completely blocked, and exclude it from further processing in the chain. We note that it is also possible that when the direct path to AP 2 is blocked, the triangle inequality may not necessarily be violated, and so while this test is conservative in the APs it excludes (thus aiding performance), it is not comprehensive in the elimination of direct path blockage scenarios. With more group of APs, the chance of detection of the blocked APs is higher. Also this scheme may fail when multiple APs are 100% blocked. However, the chance that multiple APs are blocked at the same time is quite low as the APs are usually placed at different locations, and our end-to-end evaluation suffers from these effects as and when they happen in practice. We note here that this method has very recently been applied to ToA-based ultrasound positioning [54] and we would like to apply this method to TDoA-based Wi-Fi localization in ToneTrack.



(a) No △ violation. **(b)** △ violation.

Figure 11: The classical triangle inequality can identify APs whose direct path to the client must be blocked.

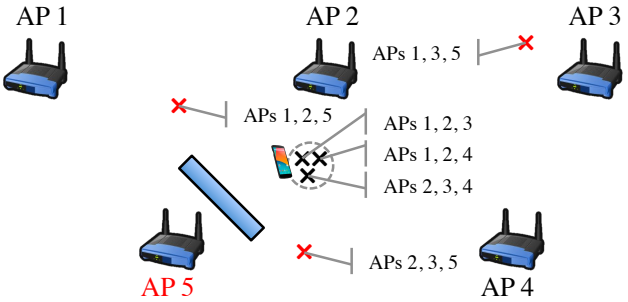


Figure 12: Employing clustering and outlier rejection to remove non-accurate estimates.

2.5.2 *Clustering and outlier rejection*

Clustering and outlier rejection further reduce the error caused by a complete blockage of the direct path signal and errors from other sources. This is based on the fact that the direct path signals of multiple APs will localize the clients close to the true source location, while reflection path signal will localize the client at random locations. As shown in Figure 12, APs 1, 2, 3 and 4 all have direct path signals while AP 5 has direct path signal blocked. Its estimates with any three APs from {1, 2, 3 and 4} will be around the true location of the mobile. A location estimate from involving AP 5 will be far away from the true location, and can be detected and removed. We can even detect the AP with direct path totally blocked. Note that we need at least four APs whose direct paths are not blocked in order to detect the blocked AP. When the number of available APs is large, the number of combinations ToneTrack needs to check can be very large. One solution to this problem is to remove some APs with small signal strength and only keep the rest for outlier rejection purposes.

2.5.3 *Final location estimation*

As noted above in Section 2.1, each pair of APs yields one TDoA estimate in the shape of a hyperbolic arc. Thus three APs are able to localize the client at the intersection of two hyperbolas.[4] Both closed form solutions and iterative algorithms can be found in [5, 31, 42]. We leverage a closed form solution in 2-D space similar to prior work [31]. With any group of three APs, we have one intersection from two hyperbolas. If we have more than four APs,

[4] If there is no intersection, we discard data from that triplet of APs.

we apply the scheme described in Section 2.5.1 and Section 2.5.2 to detect the 100% blocked AP and remove it from localization. Then we average the location estimates with all combination of three APs. When only four APs exist, the scheme described in Section 2.5.2 can not be applied. We then adopt a simple clustering algorithm to choose a group of three estimates which yields the minimum sum of distances and average them.

From ToA to TDoA. ToneTrack is based on time-difference-of-arrival (TDoA) between the mobile transmission's arrival at each pair of APs. In order to compute TDoA, ToneTrack relies on a time-synchronization mechanism between APs. This is achieved by either a wireless protocol such as SourceSync, which can achieve 5 -10 ns (95^{th} percentile) synchronization error at a typical wireless SNR ratio of 20 dB [36], or the Ethernet-based Precision Time Protocol standardized as IEEE 1588, which Broadcom has shown can provide a five nanosecond time synchronization error [4]. Other schemes include time-synchronization with light [26] and the use of distributed antenna system (DAS) [61] to bypass this time synchronization problem.

The computational load of ToneTrack is mainly a matrix multiplication of size 64 x 64. We note that with channels combined together, the matrix size is increased linearly. When there are many channels, we recommend selecting one sub-carrier out of every N adjacent sub-carriers evenly to reduce the matrix size. When there are many APs, the number of combinations for outlier rejection scheme is large which impedes ToneTrack's real-time objective. We thus only keep a limited number of APs based on the signal strengths as higher SNR presents us more accurate spectrum.

3. IMPLEMENTATION

ToneTrack is implemented on the Rice WARP platform [38] with WARPLab version 7.3. We employ a small part of the preamble of a packet which is the most robust part for our localization. For the long training symbol (LTS) in the preamble, only the middle 52 out of 64 sub-carriers are actually used. With the original LTS, only 52/64 $\times$ 20 MHz = 16.25 MHz bandwidth would be used for localization. In order to use all subcarriers, we build one symbol very similar to the LTS in 802.11 but with all the 64 subcarriers occupied. We attach this symbol just after the original LTS, incurring less than 0.1% overhead in a 1500-byte packet.

We employ five WARPs, one as the transmitter (client) and four as the receivers (APs). The carrier frequency offsets between the WARP transmitter and receivers are measured in the range of serval hundreds to several thousands of Hertz. It is much smaller than the sub-carrier size (312.5 kHz) and hence has very little effect on the ToA spectrum. So a carrier frequency offset (CFO) between mobile and AP, and pairs of APs is not a problem for ToneTrack. Each WARP kit is also attached with the FMC-RF-2X245 module to enable four radios on each board as shown in Figure 13. We connect the antennas to the WARPs with low loss LMR-400 coaxial cables. All the data recorded at the APs are retrieved through Ethernet connections between the WARPs and the server. Our super-resolution MUSIC, spectrum identification (SI), triangle inequality (TI) and clustering schemes are implemented on the server side.

AP Calibration. Due to the nonlinearity of the receiver front end across each subcarrier, we need to calibrate the channel frequency response in terms of both amplitude and phase. Note that this calibration is a one-time effort for one power-on-off cycle of the WARP. We describe our calibration steps briefly here. First, we connect the radio of the transmitter to the radio of the receiver with an RF cable. Then, we calculate the channel frequency response for each sub-carrier and calibrate the phases across each subcarrier into

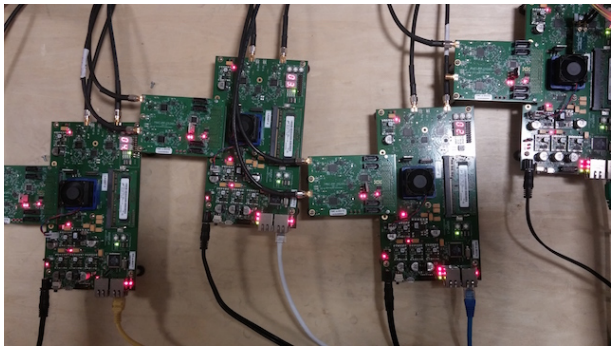


Figure 13: Each AP is a latest WARP v3 Kit with FMC-RF-2X245 module to enable 4 radios. Antennas are placed at the dedicated AP positions with low loss LMR-400 cables.

exact linear relationship with the right slope. The slope can be calculated as $2\pi\Delta f t$, where Δf is the sub-carrier size and t is the signal propagation time which can be calculated carefully by measuring the length of the cable attached between the transmitter and the receiver, adding a correction for the small extra path length caused by the splitter and the internal circuitry of the WARP radios. We also calibrate the amplitude of the frequency response across each sub-carrier to be equal. After calibration, we are able to achieve a very sharp time of arrival spectrum close to a line with only one signal transmitted through cables. This front-end linear calibration can be restricted only to ToneTrack processing. The calibration does not factor into the transmit waveform either. The calibration coefficients are calculated as phases and amplitudes for each sub-carrier and we only apply them when ToneTrack processing is called.

4. EVALUATION

To show how well ToneTrack performs in real indoor environment, we present the results from the testbed described in Section 3. First we present our evaluation methodology. Then we show our main results in Section 4.2 which answer the following:

1. What is the overall end-to-end performance with channel combination (§4.2)?
2. How much is spectrum identification scheme helping ToneTrack (§4.2.2)?
3. How does our triangle inequality scheme perform in identifying the APs with direct path totally blocked (§4.2.3)?
4. Will increasing numbers of APs improve performance (§4.2.4)?
5. What is the performance of ToneTrack with different levels of time synchronization errors between APs (§4.2.5)?

After we present our main results, we justify our choice of W_t in Section 4.3.

4.1 Experimental methodology

For our experiments, three radios on each AP is utilized to receive signals at channels 1, 5 and 9 respectively. The three radios are connected to a single antenna with combiners. The transmitter either hops across frequencies with one radio, transmitting on three channels sequentially or transmits simultaneously on all three channels with three radios. At each AP position, we collect both data traces from frequency hopping and traces from simultaneous transmissions at multiple channels. They don't have obvious performance difference. Our results presented here include all the traces.

We place the APs in a 25×20 m office, denoting them with numbers shown in Figure 14. We place clients at 40 randomly-chosen locations denoting their positions as red dots on the floor plan. 12 clients are not in the same room as the APs, with at least

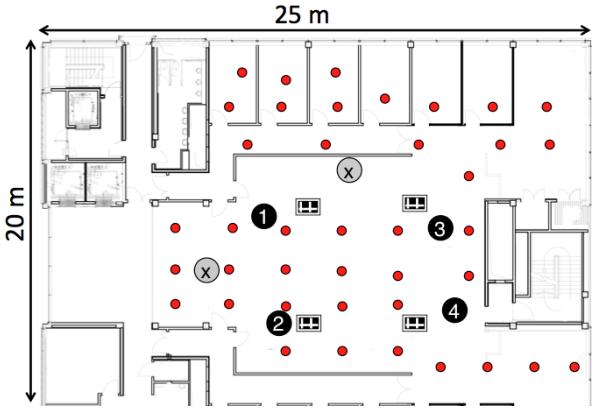


Figure 14: The indoor office environment testbed used for our experiments. The four APs used in our experiments are marked as black numbers while the client locations are marked as red dots.

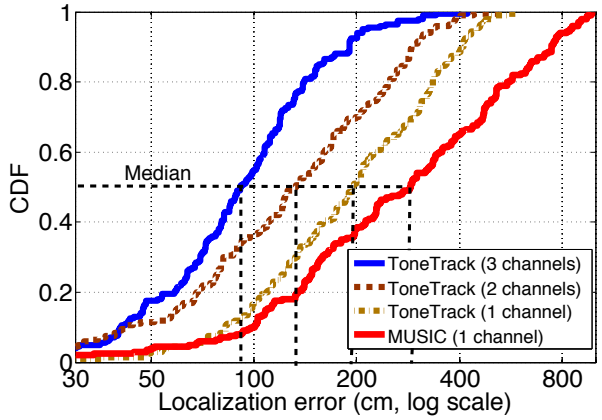


Figure 15: ToneTrack's overall localization performance with different numbers of channels and four APs.

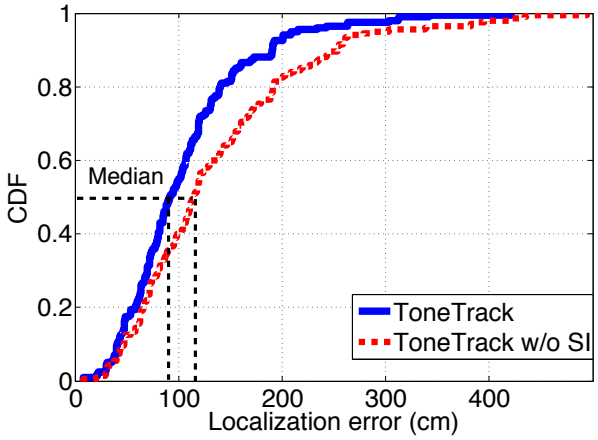


Figure 16: Isolating the effect of the spectrum identification (SI) scheme with three channels. Four APs are used in this experiment.

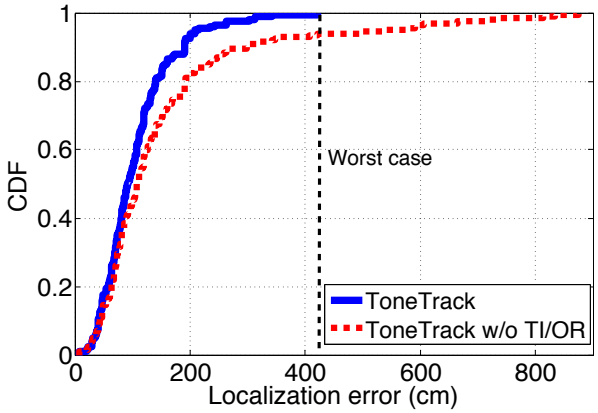


Figure 17: The effect of triangle inequality (TI) and clustering schemes.

one to two walls in between. Please note that, we only employ four APs for our main evaluation except in Section 4.2.4 where we evaluate the performance with varying number of APs.

4.2 End-to-end localization accuracy

We show the end-to-end performance evaluation of ToneTrack in this section.

4.2.1 Overall performance

The overall performance of ToneTrack is shown in Figure 15. With only three 20 MHz channels, we are able to achieve 0.9 m median accuracy in a typical office environment with strong multipaths. The median accuracies of two and one channel are 1.3 m and 1.9 m respectively, significantly better than the naïve resolution. With three channels, the 90% accuracy is around 2 m. The red curve is the CDF plot for super-resolution MUSIC without any of our proposed schemes. So even with just one channel, we are able to reduce the median localization error by 40% compared to the state-of–the-art super-resolution scheme. With our channel combination schemes applied, we further reduce the median error to below one meter which is a significant improvement with only 20 MHz channels. Also the long tail of MUSIC curve is removed in ToneTrack. We demonstrate the effectiveness of channel combination in 2.4 GHz band here with three channels. More channels can be utilized for combination at 5 GHz and 60 GHz bands which means even finer accuracy level can be achieved. We believe with the channel combination scheme proposed in ToneTrack, it's possible to achieve localization accuracy close to UWB systems.

4.2.2 Benefit of Spectrum Identification

We now isolate and show the effect of spectrum identification (SI) scheme in Figure 16. With the spectrum identification scheme, the median accuracy is improved from 116 cm to 90 cm. We can see that the spectrum identification scheme is effective in improving the performance by identifying the more accurate part of the spectrum for localization. However, we do note that when we only have three APs, we may not able to apply this scheme because discarding the inaccurate spectrum reduces the number of APs below three which is the minimum requirement for TDoA localization. However, due to the popularity of WiFi in enterprises and universities, this is not an issue as most of the time many APs can be overheard in range. We also note that this spectrum identification scheme is more effective in the environment with stronger multipaths which makes it a suitable candidate for indoor localization.

4.2.3 Impact of TI and clustering

We now remove the triangle inequality (TI) and clustering schemes to see how the performance of ToneTrack is degraded. We can see from Figure 17 that without these schemes, we have a long tail

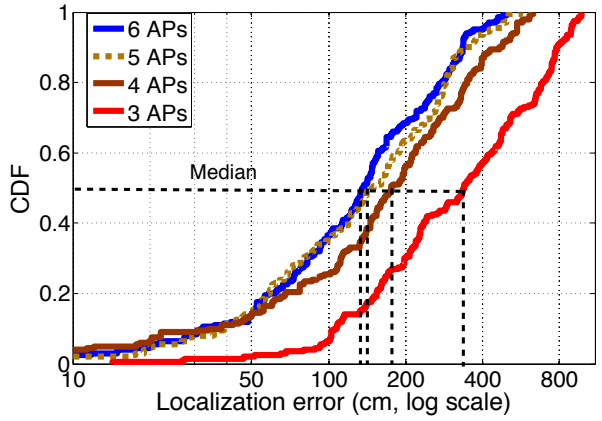


Figure 18: ToneTrack's performance with varying number of APs. Only one channel is used in this experiment.

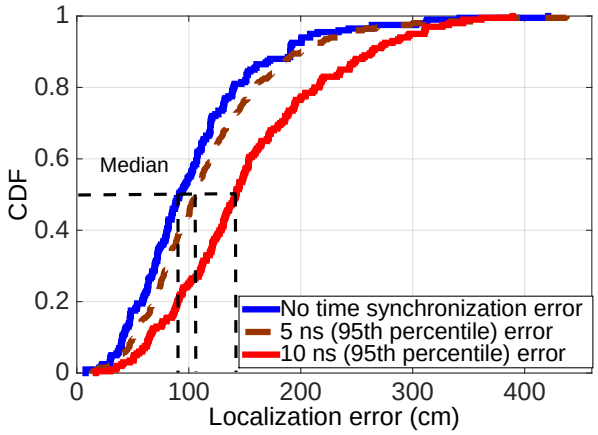


Figure 19: ToneTrack's performance with 5 ns and 10 ns (95^{th} percentile) inter-AP time-synchronization error.

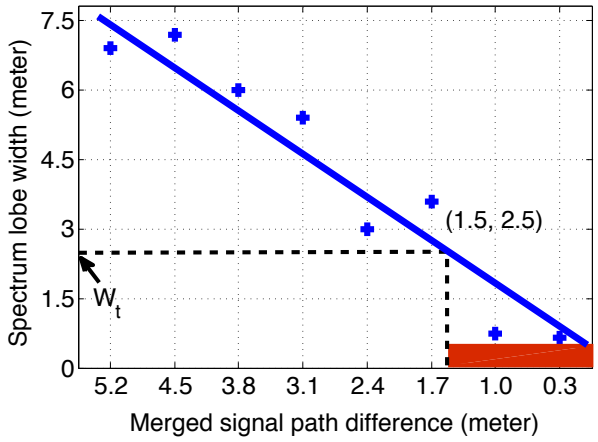


Figure 20: The merged peak width decreases when the signal path difference decreases (21 dB SNR).

on the CDF. These two schemes are effective in identifying those 'bad' APs (APs with direct path 100% blocked) and estimates with large errors. These 'bad' APs usually cause a big error because only the reflection paths exist and they localize the client to random positions. The direct-path blockage issue is more severe than multipaths. We can still try to differentiate the direct path and multipaths if both exist. With direct path 100% blocked, unless we can identify the AP and remove it from localization, it always causes a large error which significantly degrades the performance.

4.2.4 *Number of APs*

We evaluate the effect of varying number of APs on ToneTrack in this section with two more APs added at positions marked in Figure 14. In order to localize a client, we need a minimum number of three APs to have at least two hyperbolas to intersect. With only three APs, all the schemes proposed are not applied because we don't have any extra AP. From the results in Figure 18, we can see a clear gap between the CDF of three APs and four APs. With more APs added, the performance increases slightly. We believe the best solution is to identify the optimal group of APs rather than include more random APs for localization. In ToneTrack, we are able to detect the 'bad' APs whose direct path is 100% blocked and remove them. However, it's still challenging to tell which group of APs presents the best localization performance. A safe solution is to include more APs for ToneTrack. We leave the best group of APs selection problem open as our future work.

4.2.5 *Impact of synchronization error*

In our testbed, we fully synchronize all the APs. In a distributed MIMO system, there are still time synchronization errors between APs, leading to a performance degradation of ToneTrack. In order to evaluate the performance of ToneTrack with time synchronization error, we borrow the time synchronization error data from SourceSync [36] and incorporate them into our time estimates. Then we employ the new TDoA estimates to localize the clients. We can see in Figure 19, with 5 ns and 10 ns (95^{th} percentile[5]) time synchronization error, ToneTrack still performs quite well, achieving a median localization accuracy of 1.05 m and 1.4 m respectively with three channels. We expect this time synchronization error to be further reduced in the future to have an even less effect on ToneTrack's localization performance.

[5]Note that 5 ns and 10 ns are the 95th percentile values, which mean the average values are significantly smaller.

4.3 Microbenchmark: Choosing W_t

We justify our choice for the spectrum lobe width threshold W_t to differentiate the single peak and merged peak here. When more than two signals are merged or the signals are in the medium and low SNR regions, the width of the merged lobe is much larger. We show the most challenging scenario in Figure 20 where only two signals are merged and they are in the high SNR region (21 dB). With two signals in the high SNR region, the lobe width is the thinnest among the merged lobes. We show that even under these conditions, we can still choose a constant threshold value safely for a particular bandwidth with very little performance degradation. From Figure 20, we can see that the width of the merged peak is large as long as the path difference between the two signals are above 1.7 m. If we choose the threshold as 2.5 m[6] to differentiate a single and merged peak, we make mistakes only when the path difference of the two signals is below 1.5 m. Note that the merged peak position is always between the true peak positions of the two signals. When the path difference is as small as below 1.5 m, the deviation of the merged peak position from the true direct path peak position is also small. So mis-identification of the merged peak as single peak in this scenario has little effect in the performance.

[6]Note that we measure the spectrum lobe width in distance converted from time at the speed of light.

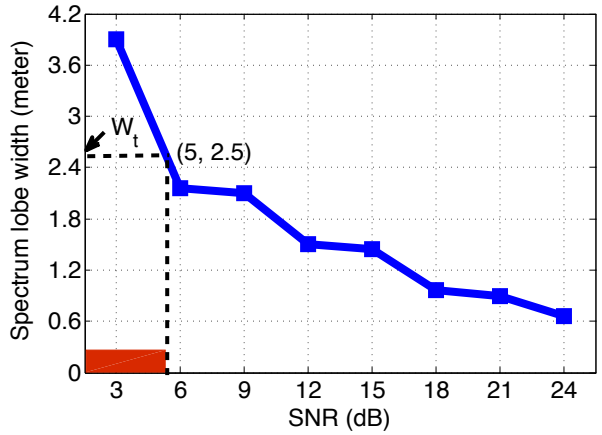


Figure 21: The lobe width of a single signal's ToA spectrum decreases when SNR increases. The lobe width increases dramatically when SNR goes below 6 dB. The red region denotes a range where ToneTrack classifies a single signal peak as a merged peak, but note the extremely low SNR.

From Figure 21, we can see clearly that the lobe width of a single signal remains well below 2.5 m as long as the SNR is above 6 dB. ToneTrack makes mistakes only in the very low SNR region (below 6 dB). In this SNR region, the accuracy level of the spectrum is anyway low and ToneTrack relies more on other APs to localize the client. So the effect of making mistakes on an inaccurate spectrum is also small. It's also noted that with many APs around, it's unlikely all the APs have low SNRs with respect to a client. We plan to adapt this threshold with varying SNR values in our future work. However, a single threshold is performing pretty well as explained.

5. RELATED WORK

Related work on indoor localization broadly groups into the following categories:

Ultrasonic and infrared based. Much early work has employed ultrasonic and infrared infrastructure. Cricket [35] employs a combination of radio and ultrasound, while Bat [14, 56] and Badge [55] leverage infrared sensors in badges carried by users.

RSSI and CSI signatures. Early work also targeted building a database of RSSI signatures at nearby APs, as RADAR [2, 3], Horus [65], and WiGEM [13] do. A slight departure from conventional approaches, Modellet [24] makes the case for a hybrid model combining fingerprint-based and model-driven localization approaches to handle data diversity and density in large scale deployments. In addition to coarser RSSI information, later work has leveraged finer channel state information (CSI). PinLoc [45] generates RF signatures to differentiate spots to within one meter, while CSITE [17] identifies attackers forging Wi-Fi management frames.

PHY-based. This newer line of work leverages various physical-layer signal processing techniques to improve accuracy. For RFID, Pinit [52] leverages antenna motion to create a synthetic aperture radar that is used to localize RFID tags, RF-Compass [51] employs RFIDs located on a robot to localize a given object and hence automatically navigate towards it, and RF-IDraw [53] traces RFID trajectories by intelligently combining various pairs of antenna spacings to yield a high degree of resolution. LTEye [20] localizes LTE clients from their uplink signal transmissions using synthetic aperture radar. Ubicarse [19] takes this one step further by making the smartphone emulate a synthetic aperture radar through user induced motion. Centaur [32] fuses RF and acoustic based ranging using a Bayesian inference framework to enable indoor localization with higher accuracy.

A recent independent work appearing in the literature with ToneTrack, Splicer [57] employs a similar idea of combining CSI information from multiple channels. While Splicer combines CSI information for a more accurate power delay profile, ToneTrack uses the combined CSI to further increase the resolution of super-resolution MUSIC for more accurate localization, integrating the CSI combination process with super-resolution time-of-arrival estimation.

AoA-based techniques. ArrayTrack [59] employs a large antenna array at the AP coupled with AoA signal processing techniques to provide fine-grained indoor localization, and the most promising approaches to indoor localization involve analyzing the signals clients send jointly: both in the spatial domain, analyzing the AoA of the signal, and in the time domain, analyzing the ToA of the signal [11, 12, 29, 44], and it is known that the two offer synergistic accuracy benefits. PinPoint [18] proposes AoA processing with cyclo-stationary analysis, and Phaser [12] combines AoA with time-of-arrival analysis. Cupid [44] reduces reflection path ambiguities by using information from smartphone's inertial sensors and human mobility. Niculescu *et al.* use mechanically-steered antennas in ad-hoc and AP networks [33].

ToneTrack falls in the category of PHY-based localization techniques, sharing some signal processing concepts with AoA-based techniques above. Since it is based on TDoA, it complements PHY- and AoA-based techniques.

ToA/TDoA/UWB-based techniques. Li and Pahlavan [25] propose using MUSIC in the frequency domain for ToA estimation. JADE [48, 49] jointly estimates angle and delay of all the arriving multipath signals. Synchronicity [60] uses MUSIC super-resolution techniques in a similar TDoA system design, but its performance is limited by radio bandwidth. Ultra-wideband radios, which have been commercialized [47], can discern sub-nanosecond time-of-flight, but performance decreases in the presence of multipath propagation and direct path blockage [10], and practical UWB radios use low data rates and power due to government regulator rules.

Compared to the above ToA and TDoA techniques, ToneTrack increases the localization granularity through a novel approach of leveraging frequency hopping to combine signals, thereby aggregating the resolution across multiple discrete channels available in a band, while not requiring a specialized ultrawideband radio.

Inertial and magnetic sensor-driven. Liu *et al.* [27] combine Wi-Fi fingerprinting with acoustic ranging between smartphone users to increase accuracy. SAIL [30] leverages fine-grained CSI information coupled with inertial dead reckoning on smartphones to enable localization with a single AP. Guoguo [28] deploys an acoustic ranging infrastructure, where the innovation is at the APs that are designed to send out acoustic beacons for ranging. Zee [37] and LiFS [63] leverage crowdsourcing and inertial dead reckoning to eliminate the calibration that RSSI fingerprinting requires. Unloc [50] uses internal landmarks with specific RF signatures to recalibrate dead reckoning schemes. SpinLoc [43] leverages the human body's attenuation of Wi-Fi signals when users spin around to estimate bearing to nearby APs.

GPS-enabled techniques. EZ [8] employs genetic algorithms to triangulate users between Wi-Fi APs coupling this with sporadic GPS fixes. COIN-GPS [34] uses GPS directly with the help of a

high gain directional antenna at the GPS receiver front-end, coupled with cloud processing to analyze longer GPS signals.

FM radio-based techniques. Another line of work leverages FM radio for indoor localization owing to its lower frequency and hence better robustness to penetration, multipath and distance of transmission. Chen *et al.* [7] leverage FM for indoor radio fingerprinting with infrastructure support, combining it with Wi-Fi. ACMI [64] uses overheard radio signals to build its fingerprinting database without even infrastructure support.

Visible light-based approaches. Epsilon [15, 23] employs visible light modulation from smart LEDs coupled with custom light sensor receivers for localization. Luxapose [21] uses off-the-shelf cameras as receivers coupled with image processing techniques. Travi-Navi [66] proposes a user-bootstrapped indoor navigation system, where followers navigate using a guider's imaging capabilities and sensor readings.

6. CONCLUSION

We have presented the design, implementation, and evaluation of ToneTrack, a TDoA-based indoor localization system that leverages the channel switches that agile radios make to increase the available bandwidth for time-based localization methods, resulting in a 90 centimeter localization accuracy across an entire office floor from four APs overhearing just three packets transmitted over three adjacent 20 MHz bandwidth channels. We have proposed a novel spectrum identification scheme to retrieve useful information from a ToA profile that is mostly inaccurate. Our proposed triangle inequality and clustering schemes also help to remove the APs when a direct path is totally blocked. ToneTrack thus pushes the envelope of localization systems in terms of their accuracy, hardware requirements, and responsiveness.

Acknowledgements

The research leading to these results has received funding from the European Research Council under the EU's Seventh Framework Programme (FP/2007-2013)/ ERC Grant Agreement no. 279976. Jie Xiong is supported by a Google European Doctoral Fellowship in Wireless Networking. We are grateful to the anonymous shepherd and reviewers whose comments helped bring the paper to its final form.

References

[1] A. Amar and A. Weiss. Fundamental resolution limits of closely spaced random signals. *IET Radar and Sonar Navigation*, 2(3):170–179, 2008.

[2] P. Bahl and V. Padmanabhan. RADAR: An in-building RF-based user location and tracking system. In *Infocom*, 2000.

[3] P. Bahl, V. Padmanabhan, and A. Balachandran. Enhancements to the RADAR user location and tracking system. Technical Report MSR-TR-2000-12, Microsoft Research, 2000.

[4] Broadcom Inc. White Paper: Ethernet time synchronization. http://www.broadcom.com/collateral/wp/StrataXGSIV-WP100-R.pdf.

[5] Y. Chan and K. Ho. A simple and efficient estimator for hyperbolic location. *IEEE Trans. on Signal Processing*, 42(8), 1994.

[6] R. Chandra, R. Mahajan, T. Moscibroda, R. Raghavendra, and P. Bahl. A case for adapting channel width in wireless networks. In *SIGCOMM*, 2008.

[7] Y. Chen, D. Lymberopoulos, J. Liu, and B. Priyantha. FM-based indoor localization. In *MobiSys*, 2012.

[8] K. Chintalapudi, A. Iyer, and V. Padmanabhan. Indoor localization without the pain. In *MobiCom*, 2010.

[9] K. Chintalapudi, B. Radunovic, V. Balan, M. Buettener, S. Yerramalli, V. Navda, and R. Ramjee. WiFi-NC: Wi-Fi over narrow channels. In *NSDI*, 2012.

[10] D. Dardari, A. Conti, U. Ferner, A. Giorgetti, and M. Win. Ranging with ultrawide bandwidth signals in multipath environments. *Proc. of the IEEE*, 97(2):404–426, Feb. 2009.

[11] G. Ding, Z. Tan, L. Zhang, Z. Zhang, and J. Zhang. Hybrid ToA/AoA cooperative localization in non-line-of-sight environments. In *VTC*, 2012.

[12] J. Gjengset, J. Xiong, G. McPhilips, and K. Jamieson. Phaser: Enabling phased array signal processing on commodity Wi-Fi access points. In *MobiCom*, 2014.

[13] A. Goswami, L. Ortiz, and S. Das. WiGEM: A learning-based approach for indoor localization. In *CoNEXT*, 2011.

[14] R. Harle and A. Hopper. Deploying and evaluating a location-aware system. In *MobiSys*, 2005.

[15] P. Hu, L. Li, C. Peng, G. Shen, and F. Zhao. Pharos: Enable physical analytics through visible light based indoor localization. In *HotNets*, 2013.

[16] IEEE Standard for Information Technology, Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications, Amendment 4: Enhancements for Very High Throughput for Operation in Bands below 6 GHz (IEEE Std 802.11ac-2013).

[17] Z. Jiang, J. Zhao, X. Li, J. Han, and W. Xi. Rejecting the Attack: Source Authentication for Wi-Fi Management Frames using CSI Information. In *Infocom*, 2013.

[18] K. Joshi, S. Hong, and S. Katti. PinPoint: Localizing Interfering Radios. In *NSDI*, 2013.

[19] S. Kumar, S. Gil, D. Katabi, and D. Rus. Accurate indoor localization with zero start-up cost. In *MobiCom*, 2014.

[20] S. Kumar, E. Hamed, D. Katabi, and L. E. Li. LTE radio analytics made easy and accessible. In *SIGCOMM*, 2014.

[21] Y.-S. Kuo, P. Pannuto, K.-J. Hsiao, and P. Dutta. Luxapose: Indoor positioning with mobile phones and visible light. In *MobiCom*, 2014.

[22] T. Laakso, V. Valimaki, M. Karjalainen, and U. Laine. Splitting the unit delay. IEEE *Sig. Proc. Mag.*, 13(1):30–60, 1996.

[23] L. Li, P. Hu, G. Shen, C. Peng, and F. Zhao. Epsilon: A visible light based positioning system. In *NSDI*, 2014.

[24] L. Li, G. Shen, C. Zhao, T. Moscibroda, J.-H. Lin, and F. Zhao. Experiencing and handling the diversity in data density and environmental locality in an indoor positioning service. In *MobiCom*, 2014.

[25] X. Li and K. Pahlavan. Super-resolution ToA estimation with diversity for indoor geolocation. *IEEE Trans. on Wireless Comms.*, 3(1), 2004.

[26] Z. Li, W. Chen, C. Li, M. Li, X. Li, and Y. Liu. Flight: Clock calibration using fluorescent lighting. In *ACM MobiCom*, 2012.

[27] H. Liu, Y. Gan, J. Yang, S. Sidhom, Y. Wang, Y. Chen, and F. Ye. Push the limit of Wi-Fi based localization for smartphones. In *MobiCom*, 2012.

[28] K. Liu, X. Liu, and X. Li. Guoguo: Enabling fine-grained indoor localization via smartphone. In *MobiSys*, 2013.

[29] Y. Luo and C. Law. Indoor positioning using UWB-IR signals in the presence of dense multipath with path

overlapping. *IEEE Trans. on Wireless Communications*, 11(10):3734–3743, Oct. 2012.
[30] A. Mariakakis, S. Sen, J. Lee, and K.-H. Kim. SAIL: Single access point-based indoor localization. In *MobiSys*, 2014.
[31] G. Mellen et al. Closed-form solution for determining emitter location using time difference of arrival measurements. *IEEE Trans. on Aerospace and Electronic Systems*, 39(3), 2003.
[32] R. Nandakumar, K. Chintalapudi, and V. Padmanabhan. Centaur: Locating devices in an office environment. In *MobiCom*, 2012.
[33] D. Niculescu and B. Nath. Ad-hoc positioning system (APS) using AoA. In *Infocom*, 2003.
[34] S. Nirjon, J. Liu, G. DeJean, B. Priyantha, Y. Jin, and T. Hart. COIN-GPS: Indoor localization from direct GPS receiving. In *MobiSys*, 2014.
[35] N. Priyantha, A. Chakraborty, and H. Balakrishnan. The Cricket location-support system. In *MobiCom*, 2000.
[36] H. Rahul, H. Hassanieh, and D. Katabi. SourceSync: A distributed wireless architecture for exploiting sender diversity. In *SIGCOMM*, 2010.
[37] A. Rai, K. Chintalapudi, V. Padmanabhan, and R. Sen. Zee: Zero-effort crowdsourcing for indoor localization. In *MobiCom*, 2012.
[38] Rice Univ. Wireless Open Access Research Platform (WARP). http://warpproject.org.
[39] T. Sarkar and O. Pereira. Using the matrix pencil method to estimate the parameters of a sum of complex exponentials. *IEEE Antenna and Propagation Magazine*, 37(1), 1995.
[40] T. Schmidl and D. Cox. Robust Frequency and Timing Synchroniation for OFDM. *IEEE Trans. on Communications*, 45(12):1613–1621, Dec. 1997.
[41] R. Schmidt. Multiple emitter location and signal parameter estimation. *IEEE Trans. on Antennas and Propagation*, AP-34(3):276–80, 1986.
[42] R. Schmidt. Least squares range difference location. *IEEE Trans. on Aerospace and Electronic Systems*, 32(1):234–242, 1996.
[43] S. Sen, R. Choudhury, and S. Nelakuditi. SpinLoc: Spin once to know your location. In *HotMobile*, 2012.
[44] S. Sen, J. Lee, K. Kim, and P. Congdon. Avoiding multipath to revive inbuilding Wi-Fi localization. In *MobiSys*, 2013.
[45] S. Sen, B. Radunovic, R. R. Choudhury, and T. Minka. You are now facing the Mona Lisa: Spot localization using phy layer information. In *MobiSys*, 2012.
[46] K. Tan, J. Fang, Y. Zhang, S. Chen, L. Shi, J. Zhang, and Y. Zhang. Fine-grained channel access in wireless LAN. In *SIGCOMM*, 2010.
[47] Time domain, inc. http://www.timedomain.com.
[48] M. Vanderveen, B. Ng, C. Papadias, and A. Paulraj. Joint angle and delay estimation (JADE) for signals in multipath environments. In *Asilomar Conf. on Signals, Systems, and Computers*, 1996.
[49] M. Vanderveen, C. Papadias, and A. Paulraj. Joint angle and delay estimation (JADE) for multipath signals arriving at an antenna array. *IEEE Communications L.*, 1(1):12–14, Jan. 1997.
[50] H. Wang, S. Sen, A. Elgohary, M. Farid, M. Youssef, and R. Choudhury. No need to war drive: Unsupervised indoor localization. In *MobiSys*, 2012.
[51] J. Wang, F. Adib, R. Knepper, D. Katabi, and D. Rus. RF-Compass: Robot object manipulation using RFIDs. In *MobiCom*, 2013.
[52] J. Wang and D. Katabi. Dude, where's my card? RFID positioning that works with multipath and non-line of sight. In *SIGCOMM*, 2013.
[53] J. Wang, D. Vasisht, and D. Katabi. RF-IDraw: Virtual touch screen in the air using RF signals. In *SIGCOMM*, 2014.
[54] Y. Wang and L. Song. An algorithmic and systematic approach for improving robustness of ToA-based localization. In *IEEE Conf. on High Performance Computing and Communications, Embedded, and Ubiquitous Computing*, 2013.
[55] R. Want, A. Hopper, V. Falcao, and J. Gibbons. The active badge location system. *ACM Trans. on Information Systems*, 10(1):91–102, Jan. 1992.
[56] A. Ward, A. Jones, and A. Hopper. A new location technique for the active office. *IEEE Personal Communications*, 4(5):42–47, Oct. 1997.
[57] Y. Xie, Z. Li, and M. Li. Precise power delay profiling with commodity Wi-Fi. In *ACM MobiCom*, 2015.
[58] J. Xiong and K. Jamieson. Towards fine-grained radio-based indoor location. In *HotMobile*, 2012.
[59] J. Xiong and K. Jamieson. ArrayTrack: A fine-grained indoor location system. In *NSDI*, 2013.
[60] J. Xiong, K. Jamieson, and K. Sundaresan. Synchronicity: Pushing the envelope of fine-grained localization with Distributed MIMO. In *HotWireless*, 2014.
[61] J. Xiong, K. Sundaresan, K. Jamieson, M. A. Khojastepour, and S. Rangarajan. Midas: Empowering 802.11ac networks with multiple-input distributed antenna systems. In *ACM CoNEXT*, 2014.
[62] L. Yang, Y. Chen, X.-Y. Li, C. Xiao, M. Li, and Y. Liu. Tagoram: Real-time tracking of mobile RFID tags to high precision using COTS devices. In *ACM MobiCom*, 2014.
[63] Z. Yang, C. Wu, and Y. Liu. Locating in fingerprint space: Wireless indoor localization with little human intervention. In *MobiCom*, 2012.
[64] S. Yoon, K. Lee, and I. Rhee. FM-based indoor localization via automatic fingerprint DB construction and matching. In *MobiSys*, 2013.
[65] M. Youssef and A. Agrawala. The Horus WLAN location determination system. In *MobiSys*, 2005.
[66] Y. Zheng, G. Shen, L. Li, C. Zhao, M. Li, and F. Zhao. Travi-Navi: Self-deployable indoor navigation system. In *MobiCom*, 2014.