SCALAR: Self-Calibrated Acoustic Ranging for Distributed Mobile Devices

Paper # 686, 12 Pages + References

ABSTRACT

In this paper, we introduce a self-calibrated acoustic ranging system that achieves sub-millimeter accuracy on distributed asynchronous mobile devices. Based on our theoretical timing model, we can precisely cancel both the system delay and the nonlinear clock drift with carefully designed OFDM ranging signals. In real-world experiments, our system achieves a ranging accuracy of $0.54\ mm$ within three meters. More importantly, our one-shot measurement scheme returns the correct distance within 0.5 seconds for cold starts without user interventions. With the new capability provided by our system, we can monitor the ambient temperature with an accuracy of 0.25 °C by measuring the slight changes in the speed of sound.

1. INTRODUCTION

Acoustic ranging has been widely used in localization and tracking for mobile devices [1]. Due to the six-order-of-magnitude difference between the speed of sound and light, acoustic ranging systems have a much higher ranging precision than radio-based systems on resource limited mobile devices. With acoustic ranging, sub-millimeter tracking accuracy has been demonstrated on commercial smartphones [2], while state-of-the-art radio-based ranging systems only provide decimeter-level accuracy [3, 4, 5, 6, 7, 8]. Furthermore, acoustic ranging systems can reuse on-device speakers and microphones that are widely available to achieve dense deployments. Therefore, it becomes an excellent candidate for high-precision localization and tracking applications, including 3D user interaction [9], motion tracking for AR/VR [2], and drone navigation [10].

One of the unsolved problems for acoustic-ranging systems is how to synchronize physically separated mobile devices without modifying the hardware. There are two major sources of timing errors in distributed audio systems. First, applications cannot precisely control the underlying system delay in audio playback and recording. Second, there are clock drifts between the transmitting and the receiving audio devices. Such timing errors introduce a *dynamic* bias in distance measurements. Most acoustic ranging systems resort to an extra calibration process to remove the timing errors [2, 11, 12]. However, the calibration process needs user intervention, *e.g.*, the user has to move the device along a given trajectory [11] or to a known position [2]. Furthermore, the calibration process assumes that the clock drifts are linear and stable. Thus, it takes a few seconds to estimate the slope of the drift but may lose synchronization within tens of minutes due to estimation errors or nonlinearity in the clock [2]. There are real-time solutions that use radio signals [1] or round-trip sound signals [13] to calibrate distributed devices on-the-fly. However, these solutions only provide coarse-grained calibration that incurs centimeter-level ranging errors [1, 13]. So, there is still a huge gap between the centimeter-level calibration accuracy and the sub-millimeter-level tracking accuracy.

In this paper, we introduce the SCALAR system, a Self-Calibrated Acoustic Ranging system that achieves sub-millimeter accuracy on distributed commercial devices. Theoretically, we prove that SCALAR perfectly cancels timing errors caused by both the system delay and the nonlinear clock drift so that we can directly measure the Time-of-Flight (ToF) between the speaker and the microphone. In real-world experiments, we show that our calibration scheme achieves sub-microsecond timing accuracy, which leads to a sub-millimeter accuracy of $0.54\ mm$ on commercial mobile devices running Android, iOS, and Linux. In our implementations, SCALAR only uses high-level audio system calls, and does not require low-level control on either the audio device or the network device. Furthermore, SCALAR does not need user intervention and outputs the correct distance within 0.5 seconds after the system starts playing sounds. This allows low duty cycle operation, *e.g.*, only operate for 0.5 seconds in every ten seconds, to save energy when performing long-term monitoring tasks.

We observe that the highly accurate and stable ToF measurements provided by SCALAR bring new opportunities for acoustic ranging. As an example, high-precision ToF can be used for monitoring the ambient temperature, since the speed of sound is physically related to the air temperature. SCALAR can measure the temperature along the sound path with an accuracy of 0.25 °C by sensing slight changes in the speed of sound when the distance between two devices is known. Compared to traditional temperature sensors, the sound-based scheme directly measures the temperature in the air instead of the temperature of the sensor. Therefore, SCALAR can detect human perceivable temperature fluctuations within a few seconds, while traditional sensors smooth the temperature at a scale of tens of seconds. The sensitivity of SCALAR allows next-generation HVAC (Heating, Ventilation, and Air Conditioning) systems to recognize different types of heat sources and take timely reactions.

We encounter three key technical challenges when developing SCALAR. The first challenge is to cancel the clock offset at a precision that is less than 10% of the audio sampling interval. For the widely supported $48\ kHz$ audio sampling rate on mobile devices, the sampling interval is $20.8\ \mu s$ (microseconds), during which the sound travels by about $7\ mm$ in distance. The sampling clock of two non-synchronized devices could be misaligned by a fraction of the sampling interval that leads to millimeter level distance errors. To achieve sub-millimeter accuracy, we use the phase of the same central frequency measured by different devices to remove the sampling clock offset. Our phase-based operation perfectly cancels the sampling misalignment, even if the sampling offset changes dynamically due to clock drifts. The second challenge is to remove the range ambiguity of phase-based measurements. As the phase measurements are limited within a range of $-\pi$ to π, we get the same phase when the device is moved by a full wavelength that converts to a phase change of 2π. To remove such ambiguity, we design an OFDM ranging signal that can measure both the coarse-grained cross-correlation distance estimation and the fine-grained phase estimation. By combining the measurements with different resolutions, we can resolve the range ambiguity within a distance of 3.4 meters using a bandwidth of just $4\ kHz$. The third challenge is to reduce the impact of the multipath effect. To address this challenge, we use the auto-correlation property of our OFDM signal to separate different sound paths. In this way, we are able to remove the impacts of objects that are more than $30\ cm$ to the Line-of-Sight (LOS) path.

Our experimental results show that SCALAR achieves a measurement accuracy of $0.54\ mm$ within a distance of three meters. Furthermore, it can perform one-shot measurement that directly returns the correct distance within $500\ ms$ of cold starts. With the new capability provided by our system, we show that we can monitor the ambient temperature with an accuracy of 0.25 °C and localize the object in a 2-D plane with an accuracy of $1.5\ mm$.

2. TIMING FOR ACOUSTIC RANGING

In this section, we first analyze the timing error sources in acoustic ranging systems. We then show that these errors can be canceled when two devices transmit and receive at the *same time* and at the *same central frequency*.

2.1 Timing Error Analysis

Timing precision is the key to acoustic ranging. There are two major sources of timing error when measuring the ToF of sound waves traveling from one device to another device.

The first source of timing error is the system delay. When playing audio, there is a system delay between the time that application puts the digital audio sample into the playback buffer and the time that the audio sample is really played out by the speaker. Similarly, there is a system delay when recording audio from the microphone. Such system delays include both the software delay of the audio drivers and the hardware delay of amplifiers and filters in the audio system. The system delay leads to a random time offset that may change from time to time when the application starts playback/recording the audio. However, once the playback/recording starts, the system delay is fixed to ensure that there is no timing distortion in the continuous flow of audio samples. In other words, the application can measure the *relative* time between two samples in a *continuous* audio flow by counting the number of samples between them.

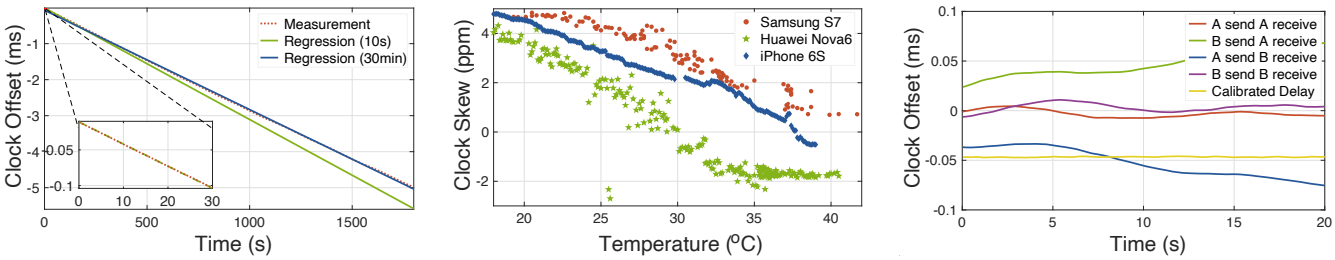
The second source of timing error is the sampling clock offset. Audio systems on mobile devices use local oscillators to generate the sampling clock. Due to hardware imperfections and temperature changes, the frequency generated by the local oscillator may have an error up to $\pm 50\ ppm$ (part-per-million) [14]. At a sampling frequency of $48\ kHz$, such error leads to a maximum frequency difference of $2.4\ Hz$ between two devices. Therefore, if device A takes $48,000$ samples in one second, device B may have $48,002.4$ samples for the same time period. With different sampling clocks, the digital samples taken by different devices are misaligned and the sampling offset between the two digital sequences keeps changing due to the accumulated clock differences.

Traditional acoustic ranging systems perform calibration every time that the audio system starts playback/recording to correct the timing error [2, 11, 15]. To estimate the system delay, the user has to put devices at known positions [2] or move them along a specific trajectory [11, 15] after the system starts. Then, the clock skew is estimated based on the linear clock offset model. Given that the time on the clock of device A is t_A, the linear module assumes the clock on device B is

$$t_B = t_A - pt - b, \tag{1}$$

where b is the initial clock offset, p is the relative clock skew (slope of the clock drift) between the two local clocks, and t is the wall clock time. For example, if there is a $50\ ppm$ clock skew, device B will be $50\ \mu s$ slower than device A for each passed second. Traditional calibration schemes require the two devices to remain static for a few seconds so that the clock skew p can be estimated. We can then compensate the clock offset using Eq. (1), by setting the initial clock offset b according to the system delay estimation.

There are several unsolved issues in the traditional calibration process. First, it requires user intervention every time that the system restarts. Second, the linear model is not accurate enough to keep long-term synchronization. Figure 1(a) shows the sampling offset between two static mobiles measured by the phase offset of Continuous Wave (CW) at a frequency of $18\ kHz$ with a sampling rate of $48\ kHz$. The clock skew is estimated with a ten-seconds time window using linear regression. While the short-term sampling offset is very close to the linear model in Eq. (1) as shown in the enlarged part in Figure 1(a), the extrapolated clock offset could be as large as $0.6\ ms$ after 30 minutes, which leads to a ranging error of $206\ mm$ when the speed of sound is $c = 343\ m/s$. Such timing error comes from



(a) Long-term linear regression errors

(b) Clock skew variations for different pairs of mobile devices

(c) Short-term nonlinear clock offsets between two mobile devices (A and B)

Figure 1: Clock offset measurement results on commercial devices

the inaccurate estimation on the clock skew p using short-term observations with a duration of ten seconds. Third, the sampling clock offset cloud be non-linear and the clock skew p may have long-term and short-term changes. In the long term, the clock offset may deviate from the linear module so that it cannot be corrected by linear regression. In the sample shown in Figure 1(a), even if we perform the linear regression over a 30-minutes period, the residual regression error is still as large as $60.5\ \mu s$, which leads to a ranging error of $20.8\ mm$. This is because the clock skew changes with the temperature so that it is not a constant in the long-term. Figure 1(b) shows the clock skew measured on three pairs of smartphones, where the transmitter and receiver are of the same brand. We heat the receiving device to change the temperature of the device and measure the temperature use a FLIR E5-XT Infrared camera [16]. The clock skews of different mobile phone pairs are all smaller than $\pm 5\ ppm$, which are much smaller than the standard requirement of $\pm 50\ ppm$ [14]. However, the clock skew significantly changes with the temperature so that linear calibrations are no longer valid when the temperature of the device changes. In the short term, Figure 1(c) shows the clock offset between two iPhone 6S, measured within ten seconds that device B starts playback/recording. We observe that clock offset in the recorded signals of device B has perceivable short-term nonlinearity. We suspect such non-linear clock offset comes from the initialization of Phase-Locked Loop (PLL) in the device. Such nonlinearity prevents accurate estimation of the clock skew within a few seconds after the audio system starts.

2.2 Timing Models

SCALAR uses the following timing model to calibrate distributed audio systems. Given a device A, we denote the time indicated by its own clock as:

$$t_A = t - q_A(t), \tag{2}$$

where t is the standard time (*e.g.*, given by GPS) and $q_A(t)$ is the clock offset function for device A when compared to the standard time. We *do not* assume that the offset function $q_A(t)$ is linear as the linear model in Eq. (1). Instead, we make the following assumption in our timing model.

Assumption: Both the system delay and the clock offset of the audio systems are quasi-static so that we can approximate them as constants within a short time period, *e.g.*, $40\ ms$, for the sound to travel from the transmitter to the receiver.

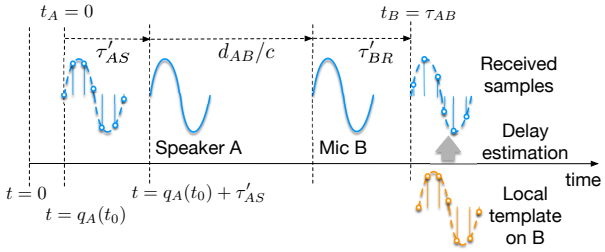


Figure 2: Delay for sound transmission and receiving.

The above assumption is valid for most commercial audio systems. First, as discussed in the previous section, the system delay should be quasi-static after the playback/recording starts so that there are no distortion in the continuous audio signal. Second, the clock skew in commercial systems are so small that the clock offset changes only by a negligible amount during a short time period. For example, with a clock skew of $5\ ppm$ as shown in Figure 1(b), the clock offset changes by $40\ ms \times 5 \times 10^{-6} = 0.2\ \mu s$ within $40\ ms$. The sound wave only travels by a negligible distance of $0.07\ mm$ within $0.2\ \mu s$. Thus, our assumption is valid for commercial mobile devices. The propagation delay of $40\ ms$ is sufficient for most indoor applications, as the sound travels by a distance of 13.6 meters within $40\ ms$. Therefore, based on our assumption, we can approximate the time of device A as a constant $t_A = t - q_A(t_0)$ during a *single* transmission that starts at t_0. Similarly, we can use $t_B = t - q_B(t_0)$ to approximate the time at the receiving device B for the same transmission.

The playback and recording delays based on our timing model are shown in Figure 2. At local time $t_A = 0$, device A starts transmitting the ranging signal by aligning the start of the ranging signal to the first playback sample in the digital audio sequence. Due to the clock offset, device A actually aligns the first sample at a standard time $t = t_A + q_A(t_0) = q_A(t_0)$. After the system delay τ'_{AS} for audio transmission on device A, the start of the signal reaches the speaker at time $t = q_A(t_0) + \tau'_{AS}$. The sound wave travels from speaker A to microphone B with a ToF of d_{AB}/c, where c is the speed of sound and d_{AB} is the distance between A and B. After a system delay, τ'_{BR}, for recording on device B, the sound

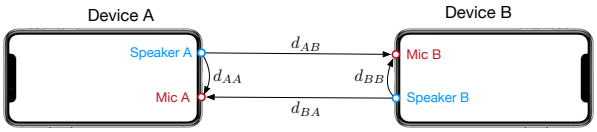


Figure 3: A pair of devices that are transmitting and receiving at the same time.

signal is received by device B as digital samples. While the samples could be misaligned, *e.g.*, the start of the received ranging signal is not aligned to a sampling point on device B as shown in Figure 2, B can still use the delay measurement scheme described in later sections to get a precise estimation of τ_{AB}. On the standard clock, the time that B receives the start of the ranging signal is $t = q_A(t_0) + \tau'_{AS} + d_{AB}/c + \tau'_{BR}$. However, as τ_{AB} is measured based on the clock on device B that is given by $t_B = t - q_B(t_0)$, we have:

$$\tau_{AB} = q_A(t_0) + \tau'_{AS} + d_{AB}/c + \tau'_{BR} - q_B(t_0). \quad (3)$$

Based on our assumption, both τ'_{AS} and $q_A(t_0)$ are constant in this short period, we can combine them and define the transmission delay as $\tau_{AS} = q_A(t_0) + \tau'_{AS}$ and drop the variable t_0 in our later discussions that only involve measurements on the same transmission instance. Similarly, we define the receiving delay as $\tau_{BR} = \tau'_{BR} - q_B(t_0)$. Thus, the measured delay is given by:

$$\tau_{AB} = \tau_{AS} + d_{AB}/c + \tau_{BR}. \quad (4)$$

Both the transmission and receiving delay τ_{AS} and τ_{BR} are unknown and slowly changing with time. However, when there are two devices that can transmit and receive at the same time, as shown in Figure 3, we can measure four delays on device A and B at the same time: τ_{AA}, τ_{AB}, τ_{BA}, and τ_{BB}, which are delays between the transmitter and the receiver specified by the subscripts. For example, τ_{AA} is the delay of the ranging signal sent by A and measured also by device A itself. In this case, we can cancel all the unknown transmission/receiving delays, given that these measurements are taken within a short time period.

THEOREM 1. *For two devices that can transmit and receive at the same time, we can calculate the distance between the two devices using four delay measurements as:*

$$d_{AB}+d_{BA} = d_{AA}+d_{BB}-c(\tau_{AA}+\tau_{BB}-\tau_{BA}-\tau_{AB}). \quad (5)$$

PROOF. We observe that when device B measures the delay of the ranging signal transmitted by device A, we have

$$\tau_{AB} = \tau_{AS} + d_{AB}/c + \tau_{BR}. \quad (6)$$

Similarly, we also have:

$$\tau_{AA} = \tau_{AS} + d_{AA}/c + \tau_{AR}, \quad (7)$$

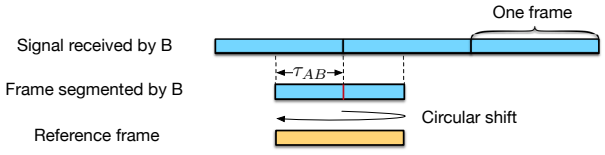
$$\tau_{BA} = \tau_{BS} + d_{BA}/c + \tau_{AR}, \quad (8)$$

$$\tau_{BB} = \tau_{BS} + d_{BB}/c + \tau_{BR}. \quad (9)$$

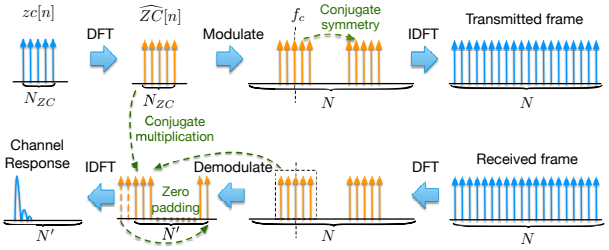
Using Eq. (6)–(9), we can cancel unknown delays by:

$$\tau_{AA} + \tau_{BB} - \tau_{BA} - \tau_{AB} = (d_{AA} + d_{BB} - d_{BA} - d_{AB})/c,$$

which directly leads to Eq. (5). □



(a) Frame structure



(b) OFDM modulation and demodulation (blue: time domain, orange: frequency domain)

Figure 4: Structure of the ranging signal.

For the symmetric setup in Figure 3, we have $d_{AB} = d_{BA}$. Furthermore, the distance between the speaker and the microphone of a given device, *e.g.*, d_{AA} or d_{BB}, is fixed and can be measured in advance. Therefore, we have:

$$d_{AB} = \frac{d_{AA} + d_{BB} - c(\tau_{AA} + \tau_{BB} - \tau_{BA} - \tau_{AB})}{2}.$$

Note that length of the front chamber of the device can also be cancelled as part of the system delays.

The derivation of Theorem 1 does not involve specific sampling rate and its result remains valid for delays that are fractions of the sampling interval. Therefore, the timing accuracy of our calibration scheme can be smaller than the sampling interval, which is 20.8 μs at 48 kHz (around 7 mm in distance). Conceptually, our scheme is similar to the round-trip calibration scheme used in the seminal BeepBeep system [13]. However, the BeepBeep solution does not consider the clock drift so that it can only cancel the system delay at the granularity of sampling intervals. Compared to the coarse-grained calibration in BeepBeep, SCALAR achieves an accuracy of about one-tenth of the sampling interval by accurately modeling the delay and canceling the clock offset on simultaneous two-way transmissions.

3. RANGING SIGNAL DESIGN

The sub-sample calibration resolution of SCALAR calls for delay measurement schemes with similar resolutions. Traditional correlation-based methods cannot fulfill this requirement, as they measure the delay by counting the number of samples [13]. Therefore, we design an OFDM ranging signal that supports phase-based delay estimation, which can directly cancel the clock offset with sub-sample accuracy.

3.1 Signal Structure

Our ranging signal consists of repeated frames as shown in Figure 4(a). We use a frame size of 40 ms, which contains 1920 samples at a sampling rate of 48 kHz.

When device B receives the signal, it first samples the signal using its own sampling clock and then segments the dig-

ital samples into frames of 1920 samples. Based on our assumption, the difference between the $40\ ms$ frame duration measured by clock of device A and B is negligible. Therefore, the signal frame received by device B is a circularly shifted version of the signal transmitted by device A, as device A repeats the frame with the same period of $40\ ms$. To ensure the signal received at the end of the frame experiences the same channel as that received at the start of the frame, we need to further assume that the coherence time T_c is larger than $40\ ms$. The coherence time is the duration that the channel remains stable and it can be estimated by $T_c \approx c/(4f_c v)$, where f_c is the carrier frequency and v is the doppler speed [17]. When the coherence time is larger than $40\ ms$, the movement speed of the devices should be smaller than $11.3\ cm/s$ for a carrier frequency of $f_c = 19\ kHz$. Therefore, we assume that the devices only moves slowly in the following discussion and this assumption is relaxed in Section 4. When the signal travels through multiple paths to reach device B, the received frame will be the linear combination of copies of the transmitted frame that are circularly shifted by different delays. We will show how to separate these multipaths in later discussions.

Our goal is to measure the amount of circular shift of the received signal at sub-sample resolution. Denote the digital sample of the frame received by B as $y[n], n = 0 \ldots, N-1$ with $N = 1920$, and its Discrete Fourier Transform (DFT) as $Y[n]$ which has the same length. In the following discussions, we use the non-capitalized/capitalized symbols to denote signals in the time/frequency domain. We denote the frame transmitted by A as $x[n]$ and its DFT as $X[n]$. By the time shifting theorem [18], if the signal is circularly shifted by a delay of τ, we have:

$$Y[n] = e^{-j2\pi n\tau f_s/N} X[n], \tag{10}$$

where f_s is the sampling frequency and j is the imaginary unit. The time shifting theorem holds for delays of a fraction of sample, because we can treat the received signal as a continuous periodical signal when considering sub-sample delays. As the transmitted signal $x[n]$ is known, we can use the phase-shift in the frequency domain to measure the delay at sub-sample resolution. The delay measurement is based on the clock of device B, which is τ_{AB} in Figure 2, .

We use the OFDM modulated Zadoff-Chu (ZC) sequence [19] to measure the phase-shift of the signal. The ZC sequence with a length of N_{zc} is given by:

$$zc[n] = e^{-j\frac{\pi u(n+1+2q)}{N_{zc}}}, n = 0, \ldots, N_{zc}-1, \tag{11}$$

where u and q are parameters of the sequence. We set q to 0, u to a coprime integer to N_{zc}, *i.e.*, $\gcd(u, N_{zc}) = 1$, and N_{zc} to an odd number. We denote the DFT of the ZC sequence as $ZC[n]$, which also has a length of N_{zc}. The ZC sequence has constant amplitude and optimal auto-correlation so that it is a good candidate for acoustic sensing tasks [20].

Modulation: Algorithm 1 shows our frequency domain modulation scheme. With a carrier frequency of f_c, a frame length of N, and a sampling frequency of f_s, the carrier frequency is at the frequency point $n_c = Nf_c/f_s$ in the frequency domain. We can carefully choose f_c and N so that n_c is an integer. For example, for a frame length of $N = 1920$ and $f_c = 19\ kHz$, we have $n_c = 760$. We rearrange the frequency points of DFT $ZC[n]$ to $\widehat{ZC}[n]$ so that the DC (zero frequency) component is at the center of the sequence, *i.e.*, $n = (N_{zc}-1)/2$. To simplify our notations, we define $h_{zc} = (N_{zc}-1)/2$ and use $2h_{zc}$ to represent $N_{zc}-1$ in the following discussions. The rearranged $\widehat{ZC}[n]$ is copied to the frequency domain frame buffer $X[n]$, with the DC component of $\widehat{ZC}[n]$ aligned to the carrier frequency n_c, as shown in Figure 4(b). We copy the conjugate of $\widehat{ZC}[n]$ to the negative frequency part of $X[n]$ so that the resulting $X[n]$ satisfies the conjugate symmetry conditions for a real-valued signal [18]. The rest parts of the frequency domain frame buffer are set to zero. After an N-point IDFT on $X[n]$, we get the real-valued time signal $x[n]$. Note that our frame length is $N = 3 \times 5 \times 128$ so that the IDFT can be calculated through a combination of radix-3, radix-5, and radix-2 Fast Fourier Transforms (FFT), which is supported by most mobile devices. The time-domain signal is a periodical signal with a period of N and a bandwidth of $f_s N_{zc}/N$.

Algorithm 1: Modulation Algorithm

Output: A modulated sequence $x[n]$ of length N
1 Generate $zc[n]$, $n = 0, \ldots, 2h_{zc}$ based on Eq. (11) ;
2 Perform N_{zc}-point DFT on $zc[n]$ to get $ZC[n]$;
3 Frequency rearrangement:
$\widehat{ZC}[n] \Leftarrow ZC[n - h_{zc}], n = h_{zc}, \ldots, 2h_{zc}$,
$\widehat{ZC}[n] \Leftarrow ZC[N_{zc} - 1 - n], n = 0, \ldots, h_{zc} - 1$;
4 OFDM modulation: $X[n] \Leftarrow 0, n = 0 \ldots N-1$,
$X[n + n_c - h_{zc}] \Leftarrow \widehat{ZC}[n], n = 0 \ldots 2h_{zc}$,
$X[n] \Leftarrow X^*[N-n], n = N/2+1, \ldots, N-1$;
5 Perform N-point IFFT on $X[n]$ to get $x[n]$;

Demodulation: The receiver first segments the received digital sequence into frames of length N and then performs an N-point DFT on each segment to get the frequency domain representation of the received signal. To demodulate, the receiver selects the N_{zc} frequency points centered at the carrier frequency f_c in the frequency domain, *i.e.*, $n = n_c - h_{zc}, \ldots, n_c + h_{zc}$. Then, these frequency points are multiplied with the conjugate of the ZC template, $\widehat{ZC}^*[n]$, where * means conjugation. We align the frequency components when performs the multiplication, with the frequency point at n_c multiplied with the DC component of $\widehat{ZC}^*[n]$, as shown in Figure 4(b). We denote the multiplication result as $\widehat{CFR}[n]$, where the DC component is at $n = h_{zc}$. Since conjugate multiplication in the frequency domain is equivalent to circular cross-convolution of two signals [18], the resulting time domain sequence will be the circular cross-correlation of the ZC sequence with the received data frame. As the auto-correlation of the ZC sequence is a perfect unit impulse function [19], we can get the complex-valued Channel Impulse Response (CIR) after converting $\widehat{CFR}[n]$ back

to the time domain by IDFT. We use spectral zero padding on $\widehat{CFR}[n]$ by inserting zeros between the positive frequency and negative frequency to expand the sequence to $N' > N_{zc}$ points. This operation is similar to the interpolation scheme used for modulation in [20]. The spectral zero padding is equivalent to time interpolating the CIR with a sinc function, *i.e.*, $\sin(x)/x$, so that the time resolution of the CIR can be improved [18].

We have verified that our frequency domain modulation/demodulation scheme is equivalent to the scheme used in [20]. However, our scheme only uses two FFT/IFFT operations per frame to demodulate the signal so that the computational cost is much smaller than the time domain demodulation scheme in [20].

3.2 Phase Measurements on OFDM Signals

We use the following theorem to measure the delay of the ranging signal.

THEOREM 2. *With the modulation/demodulation scheme in Section 3.1, if the transmitted signal is delayed by τ, the CIR will be a sinc function with the peak at $m = \tau f_s N'/N$ and the phase of the peak is given by $\varphi = -2\pi\tau f_c$.*

PROOF. In the frequency domain, the modulated signal $X[n]$ is non-zero only in the neighborhood of the central frequency, *i.e.*, $n = n_c - h_{zc}, \ldots, n_c + h_{zc}$. Using the time shifting theorem, we have:

$$\begin{aligned} & Y[n + n_c - h_{zc}] \\ = \; & e^{-j2\pi(n+n_c-h_{zc})\tau f_s/N} X[n + n_c - h_{zc}] \\ = \; & e^{-j2\pi(n+n_c-h_{zc}))\tau f_s/N} \widehat{ZC}[n], n \in [0, 2h_{zc}] \end{aligned}$$

By the property of the Zadoff-Chu sequence, we have $\widehat{ZC}[n] \times \widehat{ZC}^*[n] = 1, \forall n \in [0, 2h_{zc}]$. Therefore, we get:

$$\begin{aligned} \widehat{CFR}[n] &= \widehat{ZC}^*[n] \times Y[n + n_c - h_{zc}] \\ &= e^{-j2\pi(n+n_c-h_{zc})\tau f_s/N} \end{aligned} \quad (12)$$

for $n \in [0, 2h_{zc}]$. After zero padding and rearranging frequency components (Line 5 in Algorithm 2), we have:

$$\begin{aligned} CFR[n] &= R[n] \times e^{-j2\pi(n+n_c)\tau f_s/N} \\ &= R[n] \times e^{-j2\pi n_c \tau f_s/N} \times e^{-j2\pi n\tau f_s/N}, \end{aligned}$$

where $R[n]$ is the rectangular function centered at 0:

$$R[n] = \begin{cases} 1 & 0 \le n \le h_{zc}, \\ 1 & N' - h_{zc} \le n \le N' - 1, \\ 0 & h_{zc} < n < N' - h_{zc}. \end{cases} \quad (13)$$

We observe that the CFR is the product of three components. The first component $R[n]$ is a rectangular function that is a real-valued sinc function with peak at $n = 0$ in the time domain. The second component $e^{-j2\pi n_c \tau f_s/N}$ is a constant phase shift. Since we have $f_c = n_c f_s/N$, this phase shift is equivalent to $e^{-j2\pi\tau f_c}$. The third component $e^{-j2\pi n\tau f_s/N}$ is a phase offset that is linearly related to n. Based on the time shifting theorem, this phase offset is equivalent to a circular shift of $\tau f_s N'/N$ samples in the time domain. Therefore, the resulting time domain $cir[n]$ is a time shifted sinc function with a constant phase offset of $\varphi = -2\pi\tau f_c$. □

Algorithm 2: Demodulation Algorithm

Input: Received signal sequence $y[n]$
Output: Channel response sequence $cir[n]$ of length N' for each frame

1 Segment the received signal into frames with equal length of N;
2 **foreach** *frame $y[n]$ of length N* **do**
3 Perform N-point FFT on $y[n]$ to get $Y[n]$;
4 Conjugate multiplication: $\widehat{CFR}[n] \Leftarrow \widehat{ZC}^*[n] \times Y[n + n_c - h_{zc}], n = 0, \ldots, 2h_{zc},$;
5 Zero Padding: $CFR[n] \Leftarrow 0, n = 0 \ldots N' - 1$, $CFR[n] = \widehat{CFR}[n + h_{zc}], n = 0, \ldots, h_{zc}$, $CFR[N' - 1 - n] = \widehat{CFR}[n], n = 0, \ldots, h_{zc} - 1$;
6 Perform N'-point IFFT on $CFR[n]$ to get $cir[n]$;
7 **end**

Theorem 2 shows that delaying the signal by τ has two effects on the CIR. First, the peak of the CIR will be circularly shifted by an offset of $m = \tau f_s N'/N$ sample points. With the offset of the correlation peak, we can get the delay in terms of the number of sample points. For example, if we set $N' = N$, the offset estimation will have a coarse resolution of 7 mm at a sampling rate of 48 kHz. Second, the sinc function is real-valued with a positive peak so that the phase of the peak is equal to φ. As the wavelength of the sound wave is $\lambda_c = c/f_c$ and the delay $\tau = d/c$, we have $\varphi = -2\pi d/\lambda_c$. With phase resolution of 0.01 radians, we can measure the distance with a resolution of 0.028 mm when the sound wavelength is 18 mm.

Both the circular shift offset m and the phase φ are linear functions of the delay τ. Therefore, we can use Theorem 1 to directly cancel the unknown system delay and clock offsets in both m and φ, as shown by the following Corollaries.

COROLLARY 3.1. *Given two devices that have CIR peak offsets of m_{AA}, m_{AB}, m_{BA}, and m_{BB} for the corresponding LOS path, we have:*

$$\begin{aligned} & m_{AA} + m_{BB} - m_{AB} - m_{BA} \\ & = -\frac{f_s N'(d_{AB} + d_{BA} - d_{AA} - d_{BB})}{cN} \mod N'. \end{aligned}$$

The goal of merging the offset of the LOS path is to get a coarse-grained estimation with an error smaller than the wavelength so that we can further use the phase information to get the fine-grained sub-wavelength resolution using Corollary 3.2. Note that in Corollary 3.1, we assume that the LOS offset m are real numbers. However, the offset is measured as an integer count of samples in the CIR. To reduce the rounding errors, we over-sample the CIR by setting the N' to be four times of the frame length, *e.g.*, $N' = 4N$. This reduces the rounding error to around 1.75 mm, which is sufficient for a coarse estimation at the wavelength level, which is around 18 mm.

COROLLARY 3.2. *Given two devices that have phase measurements of φ_{AA}, φ_{AB}, φ_{BA}, and φ_{BB} on the LOS path, we have:*

$$\varphi_{AA} + \varphi_{BB} - \varphi_{AB} - \varphi_{BA} = 2\pi \frac{d_{AB} + d_{BA} - d_{AA} - d_{BB}}{\lambda_c} \mod 2\pi,$$

where λ_c is the wavelength of the central frequency f_c.

In practice, our cancellation scheme has the capability to remove the *non-linear* and *dynamic* clock offsets as predicted by our theoretical timing model. Figure 1(c) shows the delay measured by the phase of CIR for a pair of iPhone. While the non-linear clock offsets incur more than $0.042\ ms$ error in both τ_{AB} and τ_{BA}, the calibrated delay has an average error of $0.26\ \mu s$ during this period, which is equivalent to a ranging error of merely $0.09\ mm$.

Multipath effects: When the sound travels through multiple paths to reach the microphone, the resulting CIR is the linear combination of circularly shifted sinc functions with different delays, since both the modulation/demodulation operations and the channel are linear. Figure 5 shows examples of real-world CIR measurements. The LOS path corresponds to the highest peak and other small peaks are paths reflected by nearby objects, *e.g.*, the second small peak in Figure 5(d). When the LOS exists, its peak can be easily separated from multipath and the phase of the LOS path in CIR is not affected by paths that are more than $30\ cm$ away in our system, see detailed discussion on multipath in Section 4.2. Therefore, by isolating the LOS path and measuring its circular shift offset and phase, we can use Corollary 3.1 and 3.2 to derive the delay of the LOS without interference of other reflection paths.

4. SYSTEM DESIGN

In this section, we present how SCALAR coordinates multiple devices to transmit at the *same time* in the *same frequency* to enable precise cancellation of the clock offset.

4.1 OFDMA Ranging

Our Orthogonal Frequency-Division Multiple Access (OFDMA) ranging scheme allows multiple devices to send at the same central frequency so that each device can measure the offset m and phase φ of different transmitting sources using the *same* audio frame. We use a two-device system as an example, where one device acts as the master device that coordinates the calibration process. The other device, the client device, should demodulate the received audio signal and feedback the measured m and φ to the master device through other communication channels, *e.g.*, a Wi-Fi or a Bluetooth channel. Each feedback will be tagged with a timestamp so that the master can merge measurements from both devices taken within our delay constraint of $40\ ms$.

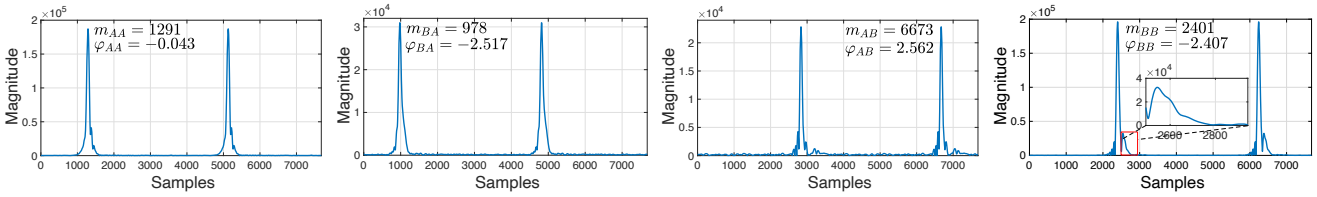
We allocate interleaved subcarriers to the master and the client as shown in Figure 6. When transmitting, the master uses only odd OFDM subcarriers and sets even subcarriers to zero. Similarly, the client uses only even subcarriers. Therefore, in a received audio frame, the receiver can separate the transmissions from the master and the client by gathering only the odd/even subcarriers. Under the OFDM parameters with a frame length of 1920 and a sampling frequency of $48\ kHz$, the frequency interval between neighboring subcarriers is $25\ Hz$. Given a clock skew that leads to a frequency offset of less than one Hertz, the interference between the neighboring subcarriers is small. To further reduce the interference between devices, we use different u values in Eq. (11) when generating the ZC sequence for the master and the client, since the cross-correlation for ZC sequences with different u values is a constant [19].

The key advantage of using interleaved subcarriers for different devices is that this scheme keeps the same ranging resolution as if all devices are using the full bandwidth. To understand this, we observe that the signal of the client device is the original full bandwidth signal (with both even and odd subcarriers) multiplied by a discrete impulse train:

$$I[n] = \begin{cases} 1 & n \bmod 2 = 0, \\ 0 & n \bmod 2 = 1, \end{cases} \tag{14}$$

in the frequency domain. Multiplication with $I[n]$ in the frequency domain is equivalent to convolution with the IDFT of $I[n]$ in the time domain. As the N'-point IDFT of $I[n]$ has only tow non-zero points at $n = 0$ and $n = N'/2$ with $I[n] = 1$ for both points, the convolution leads to two identical copies of peaks that are separated by $N'/2$ samples in the CIR. Therefore, by using only the even-subcarriers, the correlation peaks of the CIR will have the same width as the full bandwidth signal, which gives the same range resolution as using the full bandwidth. However, this scheme reduces the unambiguous range of our system by half, *e.g.*, to 6.8 meters when using a $40\ ms$ frame length, because we have two identical peaks separated by $20\ ms$ in one frame.

The master device uses odd subcarriers in the OFDM signal, where the impulse train $I[n]$ is shifted by one point in the frequency domain. This will lead to a phase shift in the time domain [18], which adds an extra phase shift of π to the second correlation peak in the CIR. In practice, the receiver cannot determine which peak aligns to the first correlation peak as there could be a time offset of more than half of a frame between the receiver and the transmitter. To solve this problem, the master device first sends in the full-bandwidth, *i.e.*, using all the subcarriers, and waits for the connection of the client device. With full-bandwidth transmissions, the CIR of the master will only have a single correlation peak which corresponds to the first peak of the odd subcarriers. After the receivers have determined the offset of the right peak, the client device will notify the master so that the master switches to the odd subcarriers. At the same time, the client starts transmitting in even subcarriers. During this short period that lasts for a few frames, both the master and the client record and process the signal continuously. Therefore, both devices can determine which correlation peak is



(a) A → A (b) B → A (c) A → B (d) B → B

Figure 5: The magnitude of CIR measurements for a pair of devices (with four combinations of sender → receiver)

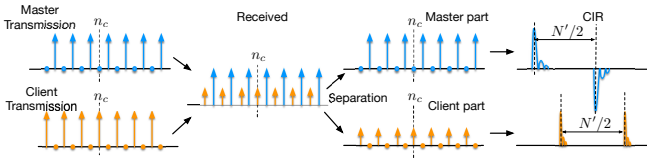


Figure 6: OFDMA ranging signal design.

the first peak for the master by comparing the measured CIR before and after the switching action. For the client signal, we can use the phase of either the first or the second peak, since both peaks have the same phase for even subcarriers.

To summarize, our ranging scheme has following steps:

1. The master device first sends in all subcarriers and waits for the client device.
2. Both the master and the client record the master transmission and determine the position of the LOS path.
3. The client notifies the master by other wireless channels.
4. The client starts transmission in even subcarriers and the master only transmits in the odd subcarriers.
5. The client measures m_{AB}, m_{BB}, φ_{AB}, and φ_{BB}, then transmits the result and timestamps to the master.
6. The master merges the result from the client, using the peak offsets to derive the coarse distance and the phases to derive the fine distance based on Corollary 3.1 and 3.2.

Figure 5 shows the CIR measurements of two devices, where device A is the master device. We can get four offsets and four phases from CIR measurements of device A and B and derive the distance using Corollary 3.1 and 3.2.

4.2 Discussions and Limitations

Our ranging scheme has the following limitations.

LOS transmission: Our system is based on the assumption that the LOS path exists. When the LOS path is blocked, the phase measurements and peak offsets may have large errors. Our system can detect whether the LOS is blocked or not, by comparing the energy of correlation peaks. When the energy of the first correlation peak is no longer larger than other peaks, we determine that the LOS path is blocked and mark the ranging results as unreliable.

Multipath mitigation: Our system can separate multipaths that are more than 30 cm longer than the LOS using the correlation property of the ZC sequence. Thus, the distance measurement is robust to surrounding objects that are more than 30 cm to the LOS. In our real-world experiments, we found that the phase measurement φ is more robust to interference than the correlation offset m. This coincides with the observations in MilliSonic [2]. Thus, most of the measurement errors comes from the correlation offset m. This leads to errors that are integer multiplications of half-wave length as shown in our experiments in Section 6. To mitigate such multipath interference, we observe that the shape of the correlation peak changes considerably when there are close multipaths. We propose to use regression algorithms based on correlation parameters, *e.g.*, the width and symmetry of the peak, to further compensate for the multipath effect. However, as our current scheme can handle most multipath conditions, we leave the study of the regression algorithm as our future work.

Fast movements: In Section 2, we assume that both devices have limited movement speeds so that the channel is coherent within a single audio frame. In real world tracking applications, the movement speed could be higher than in our assumption. In this case, the subcarriers may no longer be orthogonal to each other and the resulting CIR could be noisy. We propose to use the single frequency sinusoid solution in LLAP to track fast moving devices [21]. For example, we can mix with the OFDM signal a frequency component that is separated by more than $600Hz$ to the used subcarriers. We then use the time-domain down conversion scheme to derive the phase of the single frequency sinusoid. The resulting phase can track the device movement with a high accuracy and tolerate doppler shifts at a movement speed higher than 1 m/s. In this way, we can use the single frequency signal to track fast movements and reconcile the tracking result with the accurate range measured by the OFDM signal when the device slows down.

More than two devices: When more than two devices are used, we can measure the distances pair-by-pair in a time division multiplex manner. Moreover, we can also use the OFMDA scheme to allow more than two devices transmit at the same time. For example, for four devices, we can allocate one fourth of the subcarriers to each device, *i.e.*, device with an id of k transmits at subcarriers that has $n \mod 4 = k$. The receivers can then separate the four transmissions in the frequency domain and measure the CIR of four devices at the same time. In this case, the correlation peaks in the CIR is duplicated by four times and each duplication may have different phases. This reduces the unambiguous range to a quarter of the frame length.

Limited distance: As our system requires both devices to transmit and receive at the same time, the operational distance is limited by the transmission power of the device. For example, the speaker on a mobile phone may only al-

low us to measure distances within a few meters. However, we observe that we do not make any assumption on the relationship between τ_{BS} and τ_{BR} in Eq. (6)–(9). This means that the transmitting speaker and receiving microphone can actually be on different devices. Therefore, even if the target device only has one microphone, we can still localize it with SCALAR by treating the system as a collection of distributed speakers and microphones. When the distance between some of the speakers and microphones are known, we can solve the target distance using a series of linear equations based on our calibrated measurements. Due to space limitations, we omit detailed derivations in this paper.

5. TEMPERATURE SENSING

In this section, we discuss one of the potential applications of SCALAR: using commercial mobile devices to measure the environmental temperature.

5.1 Background

The speed of sound in the air depends on environmental variables such as temperature, humidity, and air pressure. Within the normal room temperature range, *e.g.*, $10 \sim 30$ °C, the speed of sound can be approximated as:

$$c = 331.3 + 0.606 \times T \qquad m/s, \qquad (15)$$

where the temperature T is measured in degrees Celsius (°C). While there are better approximations that relate the speed of sound to both temperature and air pressure [22], we use the approximation in Eq. (15) as it is accurate enough for our case study.

We observe that the speed of sound increases by around 0.2% when the air temperature raises by one degree Celsius. As the ToF τ_{AB} is related to the speed of sound by $\tau_{AB} = d_{AB}/c$. When the devices are separated by a constant distance of d_{AB}, the measured ToF will change with the speed of sound. As an example, for two devices that are separated by a distance of 60 cm, the ToF measurement will decrease by an amount that lead to a distance change of 1.2 mm, when the temperature raises by one degree. SCALAR is capable to detect small ToF changes at sub-millimeter level. Therefore, we can use SCALAR as a solution for temperature sensing.

5.2 Sound-based Temperature Sensing

We assume that the distance d_{AB} between the two devices is known and measure the ToF to derive the temperature. When the speed of sound is unknown, we can still use Corollary 3.1 and 3.2 to calculate the ToF. With Corollary 3.1, we can measure the delay in terms of sampling points and estimate the number of complete wave cycles along the path. We then get the fractions of wave cycles using the phase measured by Corollary 3.2. Thus, our ToF measurements can be expressed in terms of number of wave cycles along the path. We can then derive the wavelength λ_c using the known distance and the measured ToF. The speed of the sound c can then be calculated from $c = f_c\lambda_c$ and the temperate T can be derived from Eq. (15).

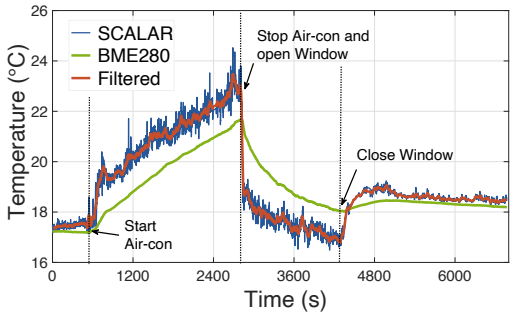


Figure 7: Temperature measurements of SCALAR

Sound-based temperature sensing is more sensitive than traditional temperature sensors such as thermistors or thermocouples. Acoustic sensing directly measures the temperature of the air, while traditional sensors measure the temperate of the probe that needs to be heated or cooled by surrounding air when the air temperature changes. Figure 7 shows the measurements of SCALAR and a Bosch BME280 temperature sensor [23] within a period of two hours. We turned on the air conditioner to heat the room at $t = 500$ seconds, turned it off and opened the window at 2760 seconds, and closed the window at 4270 seconds. For stable environments, the difference between the two temperature measurements is smaller than 0.5 °C. However, we observe that SCALAR responses to temperature changes much faster than traditional sensors. For example, when we stopped heating, SCALAR observes the temperature drop more than 6 seconds before BME280. Furthermore, the measurement of SCALAR drops by 0.5 °C within 5 seconds, while BME280 takes 30 seconds to detect the same temperature change. The low latency feedback provided by SCALAR could potentially improve the performance of control algorithms in HVAC systems.

Moreover, SCALAR can sense short-term temperature fluctuations caused by different heating devices. In a static environment, *e.g.*, with windows closed, SCALAR observes a small temperature variation of less than 0.3 °C, as shown by Figure 7. However, air conditioners and heaters introduce much larger temperature variations. Our experimental results show that SCALAR can determine whether the temperature fluctuation is caused by an air conditioner or a radiator heater by analyzing the temperature variations. Note that we carefully avoided direct air flows towards the sound path in these experiments so that these temperature fluctuations are not caused by air flows. Moreover, we can potentially reconstruct the temperature distribution in a given region by measuring multiple paths using distributed mobile devices. This could be useful for controlling the temperature in different areas of the same room or in a car.

5.3 Discussions

The user needs either the precise distance or the precise temperature to derive the other one using the measured ToF. This chicken and egg problem could be solved as follows. First, for temperature measurements, the user can use a ruler to measure the distance with an error less than one millime-

ter. When the devices are separated by 60 cm, this leads to a temperature error of less than 1 °C, which is accurate enough for most applications. Moreover, even if the user does not have a ruler, he/she can still measure the relative temperature changes by separating two devices with an unknown distance. In this case, SCALAR will use a default temperature, *e.g.*, 20 °C, to estimate the distance between devices. While the distance estimation may have up to 4% estimation error when the actual temperature is between $0 \sim 40$ °C, such error in distance only incurs 4% error in the *relative temperature changes* measured by the speed of sound. For example, when the temperature raises by 5 °C, the temperature change estimated by SCALAR will be within ± 0.2 °C of the actual change even if there is a 4% error in the initial distance estimation. The relative temperature change measurement is useful when the user wishes to increase/decrease the room temperature by a certain amount, *e.g.*, raise the temperature by 5 °C. In this case, SCALAR can monitor the temperature changes and collaborate with the HVAC system to maintain the targeting room temperature. Second, for distance measurements, the temperature error will not affect most applications. With a wrong initial temperature estimation, the distance estimation will *uniformly* increase/decrease by a small amount, *e.g.*, 4%. For tracking applications that depend on the relative movement distance, uniform changes in distance estimation will not deform the movement trace of the user.

We observe that our ToF measurements are sensitive to winds. A strong wind could potentially lead to ToF errors of more than 2%. Most of our experiments are conducted in environments where there are no human perceivable airflows. We leave the problem of mitigating the impact of winds using multiple pairs of devices as our future study.

6. EVALUATIONS

6.1 Implementation

We have implemented SCALAR on iOS, Android, and Linux systems. SCALAR operates at a central frequency of 19 kHz, occupies a bandwidth of 4 kHz (with $N_{zc} = 163$), and uses a frame length of 40 ms in our implementation. The sound signals transmitted by SCALAR are inaudible to most users. We use the same signal parameters on all platforms so that devices with different operating systems can interoperate with each other. For the iOS and Android platform, we develop stand-alone applications that perform the signal processing and measurement merging in real-time. On other platforms, we forward the signal to a desktop and process the signal in real-time in Python or MATLAB.

6.2 Evaluation on 1-D Ranging

SCALAR achieves an average 1-D ranging error of less than 0.54 mm *within a distance of 3.1 meters.* Figure 8 shows the raincloud plot of the measurement error when the devices are separated by different distances. We place a pair of Samsung S7 mobile phones at different distances on a table and measure the ground truth by a ruler. Considering the round-trip delay, the unambiguous range of the selected system parameters is around 3.4 meters so that we set the maximum distance as 3.1 meters. Note that the error is plotted in log-scale. We observe that the average error is 0.23 mm within one meter and 0.54 mm within three meters. However, there is a small amount of outliers that have a distance error of around 9 mm which increase the average ranging error when the distance is larger than one meter. This is mainly due to the error of our coarse-grained correlation that leads to an error of exactly half-wavelength when merged with the phase measurements. This type of error is also observed in other phase-based ranging systems [24]. We merge the results of four consecutive frames to reduce such outliers when the object is not moving. This reduces the number of outliers from more than 40% to 1.6% so that the impact of remaining outliers is small as shown in Figure 8.

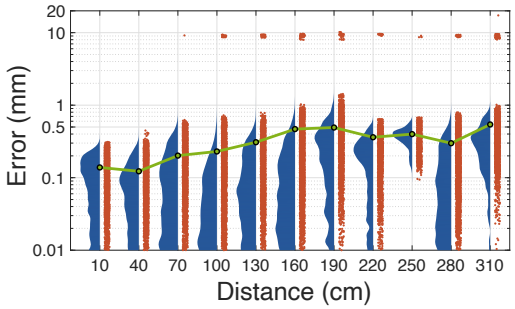
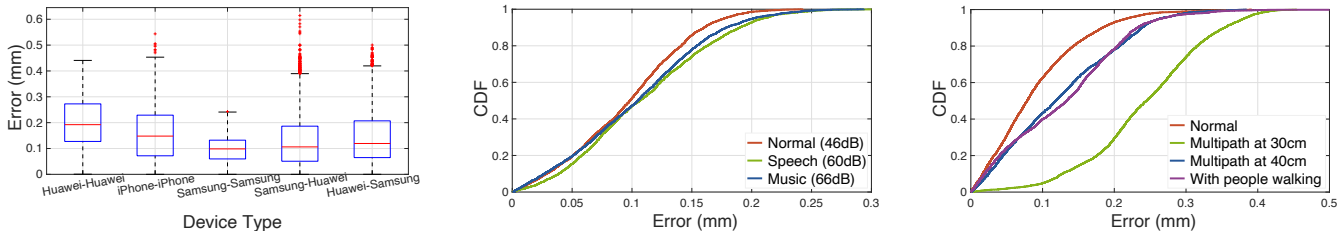


Figure 8: Ranging error distribution at different distances between devices

SCALAR achieves an average ranging error of less than 0.2 mm *for different types of devices at a distance of* 60 cm. Figure 9(a) shows the error of different types of mobile devices when placed at a distance of 60 cm, including iPhone 6S, Huawei Nova6, and Samsung S7. We observe that different types of devices has a similar performance, where the average errors are between 0.09 mm and 0.2 mm. SCALAR also works well when the two devices are from different vendors, *e.g.*, using Huawei Nova6 as the sender and Samsung S7 as the receiver.

SCALAR is robust to noises and achieves an average distance error of less than 0.15 mm *at a distance of* 60 cm *under different types of noises.* Figure 9(b) shows the performance when there are environmental noises around the devices. We observe that surrounding human speeches and musics only slightly increase the localization error from 0.097 mm to 0.114 mm and 0.105 mm, respectively.

SCALAR is robust to multipath inferences and achieves an average accuracy of less than 0.25 mm *at a distance of* 60 cm *under multipath conditions.* Figure 9(b) shows the performance under different multipath conditions. We place objects with a size of $17 \times 24 \times 5.5$ cm at a distance of 30 cm and 40 cm to the LOS path to introduce static multipaths. We also asked other users to walk around the devices at a distance of one meter to introduce dynamical multipaths. We observe that multipath only slightly increase the average error to 0.24 mm, 0.13 mm, and 0.14 mm for static multipath at 30 cm, 40 cm, and dynamic multipath conditions.



(a) Different types of devices (b) Different noise levels (c) Multipath environments

Figure 9: Robustness of distance measurements

6.3 Evaluation on 2-D Localization

For 2-D localization, we use a Samsung S7 mobile phone as the master device and a ReSpeaker 4-mic linear array on Raspberry Pi that runs Linux as the client device. We localize the master device using two microphones on the ReSpeaker that are separated by 15 cm.

SCALAR can localize targets in the 2-D plane with an average accuracy of 1.71 mm. Figure 10(a) shows the localization error in a 30×30 cm square with the center at 60 cm to the microphone array. The ground truth locations are marked on the table as a grid. We repeat the measurement for 2000 times for each grid point and the measured results are shown as red points in the figure. Note that our localization is performed by directly placing the device on grid points without moving the device or using historical traces.

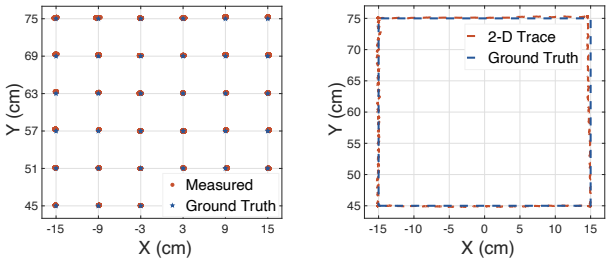
SCALAR can track moving objects in the 2-D plane with an accuracy of 1.45 mm. Figure 10(b) shows the trace of moving the mobile phone on a square with an average speed of 1.6 cm/s, which is smaller than the Doppler speed limit as discussed in Section 4.2. The average tracking error is 1.45 mm and the maximum error is 5.2 mm.

6.4 Evaluation on Temperature Sensing

To evaluate the temperature sensing capability of SCALAR, we separate two Samsung S7 by a fixed distance of 60 cm on a table and measure the temperature as described in Section 5. If not specified, we keep the doors and windows of the room closed and avoid any human perceivable air flows in the testing area.

SCALAR achieves an average temperature accuracy of less than 0.25 °*C in static environments when compared to commercial temperature sensors.* Figure 11(a) shows the error distribution of SCALAR, where the ground truth is provided by BME280, which has a temperature resolution of 0.01 °C and absolute accuracy of 0.5 °C [23]. We repeat the experiments for more than 6,000 times in a period of three months by restarting SCALAR in different indoor environments, including labs, meeting rooms, and apartments with room temperatures in the range of $7 \sim 28$ °C. The average temperature error of SCALAR is smaller than 0.25 °C and the maximum error is less than 0.9 °C under different room temperatures. Figure 11(b) shows that SCALAR can reliably measure the temperature when the devices are separated by 30 cm to 150 cm with room temperatures of $14 \sim 16$ °C.

SCALAR is sensitive enough to recognize different types of heat sources with an accuracy higher than 90%. We collected temperature measurements for three different scenarios: normal room without heating, heated by air conditioners and radiator heater, with monitoring duration of 166.7, 221.4, and 292.0 minutes, respectively. We observe that the normal indoor environment has a small temperature standard deviation of 0.10 °C within a time window of 30 seconds, but air conditioners and heaters introduce higher variations of 0.28 °C and 0.18 °C. While both are heating devices, the temperature variation patterns of the air conditioner and the heater are also different. Figure 11(c) shows the amplitude of low-frequency and high-frequency variation in different environments obtained by performing FFT over time windows with a length of 100 seconds. We observe that different types of heating devices are clearly separable using the spectrum of temperature variations. With a decision tree algorithm based on the spectrum features, we can recognize the normal environment, the air conditioner, and the radiator heater with an accuracy of 96.0%, 91.8%, and 92.8%, respectively.

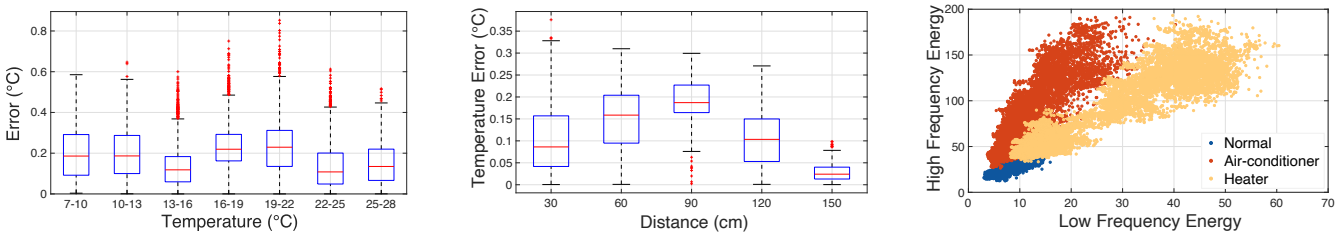


(a) 2-D localization accuracy (b) Tracking example

Figure 10: Performance of 2-D localization and tracking

6.5 System Performance

SCALAR can process the audio signal in real-time on commercial iOS and Android mobile phones. Table 1 shows the time for SCALAR to process one audio frame with a duration of 40 ms on different devices. The signal processing delay includes the procedures of segmenting the raw signal, demodulation, and extracting the peak offsets and phases of both the master and the client device. The fusion delay is the process for the master device to merge the measurements and output the distance. With the vDSP acceleration framework on iOS, SCALAR incurs negligible computational cost with a CPU occupation of less than 0.5% on iPhone 6S. Our Java-based implementation on Android also meets the real-time processing requirements, but takes more than 50 times longer time than the efficient vDSP implementation.



(a) Temperature measurement accuracy

(b) Temperature measurement accuracy for different distances

(c) Temperature variation spectrum distribution in different environments

Figure 11: Temperature sensing evaluations

Table 1: Processing time on different devices

Device	Signal processing (ms)	Fusion (ms)
iPhone 6S	0.185±0.01	0.0014±0.0001
Samsung S7	8.654±1.995	0.0082±0.0034

SCALAR returns the accurate measurement within 411 ms *in cold start.* We repeated the cold start process for more than one hundred times on a pair of Samsung S7. On average, the audio signals take 2.94 frames (118 ms) to stabilize for reliable offset and phase measurements. Furthermore, it takes 293 ms for the client to notify the master to switch to the odd subcarriers through Wi-Fi. This gives an overall cold start latency of less than 411 ms.

SCALAR incurs an overall power consumption of less than 500 mW *on the Android platform.* We use Powertutor [25] to measure the power consumption of SCALAR on Samsung S7. The average power consumption for CPU and Audio are 115.6 mW and 384.3 mW when SCALAR is operating continuously.

7. RELATED WORK

Recent works related to SCALAR are in the following four categories.

Device-free acoustic ranging: Device-free acoustic ranging systems use sound signals reflected by a moving object to track the target. As signals are transmitted and received by the same device and these systems, they are synchronized by the hardware and no calibration is needed [26, 21, 27]. Synchronized device-free systems have been widely used for gesture recognition [26, 28, 29], vital sign monitoring [30, 31, 32], and localization [33, 34]. In addition to in-air acoustic propagation, synchronized systems can also measure the structure-borne sound to localize objects on solid surfaces with centimeter-level accuracy [35, 36, 37, 20]. As physical connections are required for synchronized operation, these systems cannot be deployed in a distributed way.

Device-based acoustic ranging: Device-based ranging systems localizes a target device that is actively transmitting/receiving sound waves [15, 9, 38]. Most device-based ranging systems use the phase-based approach to improve tracking accuracy. CAT develops a distributed FMCW system to achieve a tracking accuracy of 5 mm [11]. Vernier uses an efficient phase-change estimation algorithm to track a moving device with an accuracy of 4 mm [12, 39]. MilliSonic achieves sub-millimeter tracking accuracy by combining the correlation-based and phase-based ranging [2]. However, these acoustic ranging systems require a calibration process to remove the unknown clock offset [2, 12, 39, 11] or have to use synchronized sources to localize the target [40, 41, 9]. This additional calibration requirement limits their applications to tracking the relative movement instead of measuring the true distance between devices [42, 10].

Calibration schemes: To synchronize distributed devices, Cricket uses Radio Frequency (RF) transmissions to calibrate the audio system [1]. BeepBeep measures round-trip delay to reduce the impact of system delay [13]. However, these calibration schemes do not consider the sampling clock drift. Thus, their accuracy is limited at the centimeter-level [13, 43], which is an order-of-magnitude larger than the millimeter-level tracking accuracy achieved by phase-based schemes. Phase-based calibration has been applied in RF-based ranging to achieve centimeter-level accuracy [24]. However, such systems are susceptible to multipath effects as they measure a single frequency at a time.

Temperature measurements: Acoustic-based systems can remotely measure the air temperature, so they are suitable for measuring temperatures of inaccessible locations [44, 45], *e.g.*, the temperature in a reaction chamber. With multiple transmitters and receivers, it is also possible to rebuild the temperature distribution in a given region [46, 47]. However, most of existing systems use specialized and synchronized transceivers so that these approaches cannot be applied to commercial mobile devices. Recently, there is an increased interest in mobile-based environmental temperature monitoring [48, 49]. SST uses the Time Difference of Arrival (TDoA) of chirp signals received by two synchronized microphones on the same mobile phone to achieve an accuracy of 0.5 °C [48]. However, due to the short distance between microphones on the same device, the synchronized solution in SST should use audible sounds with a large bandwidth of 20 kHz to achieve the desired accuracy.

8. CONCLUSIONS

In this paper, we developed SCALAR, a fine-grained calibration scheme for acoustic ranging systems on distributed mobile devices. Our solution uses existing low-cost hardware on mobile devices to achieve repeatable sub-millimeter level ranging accuracy. We envision SCALAR will enable new mobile applications that require highly accurate and stable range information in the near future.

9. REFERENCES

[1] Nissanka B Priyantha, Anit Chakraborty, and Hari Balakrishnan. The cricket location-support system. In *Proceedings of ACM MobiCom*, 2000.

[2] Anran Wang and Shyamnath Gollakota. MilliSonic: Pushing the limits of acoustic motion tracking. In *Proceedings of ACM CHI*, 2019.

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

[4] Deepak Vasisht, Swarun Kumar, and Dina Katabi. Decimeter-level localization with a single WiFi access point. In *Proceedings of Usenix NSDI*, 2016.

[5] Jie Xiong and Kyle Jamieson. ArrayTrack: A fine-grained indoor location system. In *Proceedings of Usenix NSDI*, 2013.

[6] Roshan Ayyalasomayajula, Deepak Vasisht, and Dinesh Bharadia. BLoc: CSI-based accurate localization for BLE tags. In *Proceedings of ACM CoNEXT*, 2018.

[7] Yaxiong Xie, Jie Xiong, Mo Li, and Kyle Jamieson. mD-Track: Leveraging multi-dimensionality for passive indoor Wi-Fi tracking. In *Proceedings of ACM MobiCom*, 2019.

[8] Han Ding, Jinsong Han, Chen Qian, Fu Xiao, Ge Wang, Nan Yang, Wei Xi, and Jian Xiao. Trio: Utilizing tag interference for refined localization of passive RFID. In *Proceedings of IEEE INFOCOM*, 2018.

[9] Cheng Zhang, Qiuyue Xue, Anandghan Waghmare, Sumeet Jain, Yiming Pu, Sinan Hersek, Kent Lyons, Kenneth A Cunefare, Omer T Inan, and Gregory D Abowd. Soundtrak: Continuous 3D tracking of a finger using active acoustics. In *Proceedings of ACM UbiComp*, 2017.

[10] Wenguang Mao, Zaiwei Zhang, Lili Qiu, Jian He, Yuchen Cui, and Sangki Yun. Indoor follow me drone. In *Proceedings of ACM MobiSys*, 2017.

[11] Wenguang Mao, Jian He, and Lili Qiu. CAT: high-precision acoustic motion tracking. In *Proceedings of ACM MobiCom*, 2016.

[12] Yunting Zhang, Jiliang Wang, Weiyi Wang, Zhao Wang, and Yunhao Liu. Vernier: Accurate and fast acoustic motion tracking using mobile devices. In *Proceedings of IEEE INFOCOM*, 2018.

[13] Chunyi Peng, Guobin Shen, Yongguang Zhang, Yanlin Li, and Kun Tan. BeepBeep: a high accuracy acoustic ranging system using COTS mobile devices. In *Proceedings of ACM SenSys*, 2007.

[14] International Electrotechnical Commission. Digital audio interface – part1: General. IEC 60958-1, 2014.

[15] Sangki Yun, Yi-Chao Chen, and Lili Qiu. Turning a mobile device into a mouse in the air. In *Proceedings of ACM MobiSys*, 2015.

[16] FLIR E5-XT infraed camera. https://www.flir.com/products/e5-xt/, 2015.

[17] David Tse and Pramod Viswanath. *Fundamentals of wireless communication*. Cambridge university press, 2005.

[18] Alan V. Oppenheim, Alan S. Willsky, and S. Hamid. *Signals and Systems*. Pearson, 1996.

[19] Branislav M Popovic. Generalized chirp-like polyphase sequences with optimum correlation properties. *IEEE Transactions on Information Theory*, 38(4):1406–1409, 1992.

[20] Ke Sun, Ting Zhao, Wei Wang, and Lei Xie. VSkin: Sensing touch gestures on surfaces of mobile devices using acoustic signals. In *Proceedings of ACM MobiCom*, 2018.

[21] Wei Wang, Alex X. Liu, and Ke Sun. Device-free gesture tracking using acoustic signals. In *Proceedings of ACM MobiCom*, 2016.

[22] Lawrence E Kinsler, Austin R Frey, Alan B Coppens, and James V Sanders. *Fundamentals of Acoustics*. Wiley, fourth edition edition, 2000.

[23] Bosch. BME280 combined humidity, pressure and temperature sensor, 2018.

[24] Miklós Maróti, Péter Völgyesi, Sebestyén Dóra, Branislav Kusỳ, András Nádas, Ákos Lédeczi, György Balogh, and Károly Molnár. Radio interferometric geolocation. In *Proceedings of ACM SenSys*, 2005.

[25] Lide Zhang, Birjodh Tiwana, Zhiyun Qian, Zhaoguang Wang, Robert P. Dick, Zhuoqing Morley Mao, and Lei Yang. Accurate online power estimation and automatic battery behavior based power model generation for smartphones. In *Proceedings of IEEE CODES+ISSS*, 2010.

[26] Rajalakshmi Nandakumar, Vikram Iyer, Desney Tan, and Shyamnath Gollakota. FingerIO: Using active sonar for fine-grained finger tracking. In *Proceedings of ACM CHI*, 2016.

[27] Sangki Yun, Yi-Chao Chen, Huihuang Zheng, Lili Qiu, and Wenguang Mao. Strata: Fine-grained acoustic-based device-free tracking. In *Proceedings of ACM MobiSys*, 2017.

[28] Wenjie Ruan, Quan Z Sheng, Lei Yang, Tao Gu, Peipei Xu, and Longfei Shangguan. AudioGest: enabling fine-grained hand gesture detection by decoding echo signal. In *Proceedings of ACM Ubicomp*, 2016.

[29] Ke Sun, Wei Wang, Alex X. Liu, and Haipeng Dai. Depth aware finger tapping on virutal displays. In *Proceedings of ACM MobiSys*, 2018.

[30] Rajalakshmi Nandakumar, Shyamnath Gollakota, and Nathaniel Watson. Contactless sleep apnea detection on smartphones. In *Proceedings of ACM MobiSys*, 2015.

[31] Tianben Wang, Daqing Zhang, Yuanqing Zheng, Tao Gu, Xingshe Zhou, and Bernadette Dorizzi. C-FMCW

based contactless respiration detection using acoustic signal. In *Proceedings of ACM UbiComp*, 2018.

[32] Anran Wang, Jacob E Sunshine, and Shyamnath Gollakota. Contactless infant monitoring using white noise. In *Proceedings of ACM MobiCom*, 2019.

[33] Bing Zhou, Mohammed Elbadry, Ruipeng Gao, and Fan Ye. BatMapper: Acoustic sensing based indoor floor plan construction using smartphones. In *Proceedings of ACM MobiSys*, 2017.

[34] Yu-Chih Tung and Kang G Shin. Echotag: Accurate infrastructure-free indoor location tagging with smartphones. In *Proceedings of ACM MobiCom*, 2015.

[35] Yu-Chih Tung and Kang G Shin. Expansion of human-phone interface by sensing structure-borne sound propagation. In *Proceedings of ACM MobiSys*, 2016.

[36] Jian Liu, Chen Wang, Yingying Chen, and Nitesh Saxena. VibWrite: Towards finger-input authentication on ubiquitous surfaces via physical vibration. In *Proceedings of ACM CCS*, 2017.

[37] Jian Liu, Yingying Chen, Marco Gruteser, and Yan Wang. VibSense: Sensing touches on ubiquitous surfaces through vibration. In *Proceedings of IEEE SECON*, 2017.

[38] Zengbin Zhang, David Chu, Xiaomeng Chen, and Thomas Moscibroda. Swordfight: Enabling a new class of phone-to-phone action games on commodity phones. In *Proceedings of ACM MobiSys*, 2012.

[39] Yunhao Liu, Jiliang Wang, Yunting Zhang, Linsong Cheng, Weiyi Wang, Zhao Wang, Weimin Xu, and Zhenjiang Li. Vernier: Accurate and fast acoustic motion tracking using mobile devices. *IEEE Transactions on Mobile Computing*, 2019.

[40] Qiongzheng Lin, Zhenlin An, and Lei Yang. Rebooting ultrasonic positioning systems for ultrasound-incapable smart devices. In *Proceedings of ACM MobiCom*, 2019.

[41] Jie Yang, Simon Sidhom, Gayathri Chandrasekaran, Tam Vu, Hongbo Liu, Nicolae Cecan, Yingying Chen, Marco Gruteser, and Richard P. Martin. Detecting driver phone use leveraging car speakers. In *Proceedings of ACM MobiCom*, 2011.

[42] Huanle Zhang, Wan Du, Pengfei Zhou, Mo Li, and Prasant Mohapatra. DopEnc: acoustic-based encounter profiling using smartphones. In *Proceedings of ACM MobiCom*, 2016.

[43] Patrick Lazik, Niranjini Rajagopal, Bruno Sinopoli, and Anthony Rowe. Ultrasonic time synchronization and ranging on smartphones. In *Proceedings of IEEE RTAS*, 2015.

[44] Wen-Yuan Tsai, Hsin-Chieh Chen, and Teh-Lu Liao. High accuracy ultrasonic air temperature measurement using multi-frequency continuous wave. *Sensors and Actuators A: Physical*, 132(2):526–532, 2006.

[45] YS Huang and Ming-Shing Young. An accurate ultrasonic distance measurement system with self temperature compensation. *Instrumentation Science and Technology*, 37(1):124–133, 2009.

[46] Xuehua Shen, Qingyu Xiong, Xin Shi, Kai Wang, Shan Liang, and Min Gao. Ultrasonic temperature distribution reconstruction for circular area based on markov radial basis approximation and singular value decomposition. *Ultrasonics*, 62:174–185, 2015.

[47] Ruixi Jia, Qingyu Xiong, Guangyu Xu, Kai Wang, and Shan Liang. A method for two-dimensional temperature field distribution reconstruction. *Applied Thermal Engineering*, 111:961–967, 2017.

[48] Chao Cai, Henglin Pu, Menglan Hu, Rong Zheng, and Jun Luo. SST: Software sonic thermometer on acoustic-enabled IoT devices. *IEEE Transactions on Mobile Computing*, 2020.

[49] Joseph Breda, Amee Trivedi, Chulabhaya Wijesundara, Phuthipong Bovornkeeratiroj, David Irwin, Prashant Shenoy, and Jay Taneja. Hot or Not: Leveraging mobile devices for ubiquitous temperature sensing. In *Proceedings of ACM BuildSys*, 2019.