UbiTrack: Enabling Scalable & Low-Cost Device Localization with Onboard WiFi

Wenpeng Wang, Zetian Liu, Jiechao Gao, Nurani Saoda, Bradford Campbell
University of Virginia
wangwp,zl4dc,jg5ycn,saoda,bradjc@virginia.edu

ABSTRACT

Wireless sensing and the Internet of Things support real-time monitoring and data-driven control of the built environment, enabling more sustainable and responsive infrastructure. As buildings and physical structures tend to be large and complex, instrumenting them to support a wide range of applications often requires numerous sensors distributed over a large area. One impediment to this type of large-scale sensing is simply tracking where exactly devices are over time, as the physical infrastructure is updated and interacted with over time. Having low-cost but accurate localization for devices (instead of users) would enable scalable IoT network management, but current localization approaches do not provide a suitable tradeoff in terms of cost, energy, and accuracy for low power devices in unknown environments.

We propose UbiTrack, a low-cost indoor positioning system that enables accurate tracking for single antenna commodity WiFi devices, without the need for a complex antenna array. UbiTrack leverages two-way channel state information (CSI) across all WiFi channels to measure the distance between nodes, and uses a new probabilistic localization algorithm based on Bayesian estimation to locate each device. We demonstrate the system on commodity $4.00 ESP32 WiFi chips and realize 1-meter level position accuracy in an indoor environment. This approach provides localization for everyday IoT devices, enabling more scalable deployments and new IoT applications.

CCS CONCEPTS

• **Computer systems organization** → **Embedded and cyber-physical systems**.

KEYWORDS

Indoor localization, smart IoT devices, WiFi channel state information (CSI), Bayesian estimation

1 INTRODUCTION

New sensors, more computing power, reliable mobile connectivity, and easier production is enabling growth in the number of deployed Internet of Things (IoT) devices. In turn, this availability and cost-effectiveness will continue to make new sensor applications possible, including large-scale monitoring and detection. Many applications will require high density deployment of IoT devices, creating a new challenge to determine and track the location of every deployed device. Overcoming this challenge will require automatic localization for devices. Automatic device localization will also simplify device installation, enable smart building deployments to adapt to changes in building use and physical structure over time, and unlock the potential for numerous location-based applications. However, many of today's IoT nodes do not include any indoor localization capabilities, and thus require periodic manual location checks and label updates. This limits the scalability of the smart IoT system in buildings.

Despite significant progress in the general field of localizing people, accurately localizing small commodity IoT devices presents a different challenge. Many approaches have a significant drawback, either providing low accuracy with multiple meter error, requiring uncommon onboard hardware or radios, or requiring significant infrastructure to be present. Yet, enabling accurate location-based services for small IoT devices could have many benefits. For example, hospitals could easily keep track of their mobile and shared equipment to more effectively manage resources, networks in homes could identify and block malicious devices accessing the network outside building walls, and small items like keys could be easily found when lost.

Enabling localization services for many resource-constrained devices on a large scale, rather than localization for human users or on an ad-hoc basis presents several design challenges that must be considered: (1) The additional cost for localization must be minimized. Using expensive hardware limits the scalability of the system. The design should leverage existing hardware components and avoid adding extra hardware. (2) Many IoT devices have a restricted energy budget. The localization approach cannot require a large number of measurements. (3) The localization accuracy must be sufficient. As devices and sensors have contextual clues (e.g. a door sensor is likely on a door) or are attached to physical objects (e.g. a chair monitoring sensor), centimeter-scale accuracy is not required, however, meter-level accuracy is needed to resolve ambiguities and track devices inside of a building.

There exist various works for providing indoor localization for devices [16, 22, 29, 33]. However, they either do not consider hardware limitations, are too energy-expensive, or do not meet the accuracy requirements, making them unsuitable for a dense indoor IoT network. Specifically, some works use ultra-wideband (UWB)

radios or software defined radio (e.g. a USRP) that permit using a large bandwidth and can calculate the time of flight (ToF) quite accurately [3, 17]. However, these approaches require dense deployments (as UWB signals do not travel through walls well) or expensive USRP hardware (>$1000 each), which is not applicable to IoT systems which need to be cost-effective to be largely deployed. Some approaches identify the cost problem and develop indoor localization systems using ubiquitous radios like WiFi [12] or Bluetooth [25]. However, these approaches either require using an array of antennas (3 to 12), require NICs that are expensive (i.e. the Intel 5300 at $30), or are high power and not practical for IoT systems where energy is limited and most deployed nodes cost less than the NIC card itself.

In this paper, we target the imbalance between scalability, energy, and accuracy, and propose UbiTrack, a network localization system that enables scalable localization for single antenna WiFi devices. Instead of using special radios like UWB or mmWave radar, UbiTrack leverages commodity WiFi radios already found in today's IoT devices. UbiTrack leverages the Channel State Information (CSI) from WiFi signals to calculate distance between nodes. We also designed a new communication method between nodes that enable the ad-hoc and association-less wifi connection between devices, this removes the need to go through re-association process every time a nodes wakes up and communicate with nearby nodes, reduces the energy cost of UbiTrack and helps prolong the operating time of the deployed device. Further, we design the communication between nodes to be ad-hoc and association-less, which reduces the total energy cost to perform ranging. To make our localization system more accurate and reduce redundant ranging to preserve energy, we design a novel positioning algorithm based on our proposed Historical-Bayesian (HiBay) estimation.

There are also a Fine Time Measurement (FTM) protocol supported on newer 802.11mc protocol, however FTM is now only available on a few platforms, and is not supported on any of the existing deployed devices, not to mention the raw FTM measurement have huge location-dependent errors that can sometimes significantly degrade the localization performance, especially on the crowded 2.4 GHz bands [10]. Other CSI-based approach uses the desktop-class chips like Intel 5300, leveraging an antenna array and 5 GHz for better resolution. Their approaches are only suited to localize targets that use 5GHz WiFi band, such as mobile phone or tablets, and the performance would be significantly degraded if used in 2.4 GHz band. Instead, our approach uses a different method for ranging which leverages all the communication channels on a single antenna on 2.4 GHz, for the fact that most WiFi IoT devices are working on this band instead of 5 GHz. To demonstrate the feasibility of our system, we implement a prototype of UbiTrack using the cost-effective Espressif ESP32 WiFi chip [8]. Compared to the Intel 5300, the ESP32 is an order of magnitude cheaper ($4 compared to $30), more energy-efficient (battery-powered vs wall-powered), and already ubiquitous. Considering the market share of ESP32 chips (more than 100 million have been sold), our approach can directly extend to numerous existing IoT networks without additional cost. We also believe that enable location services will have enough value to add an ESP32 chip to an IoT device as just a localization module a viable prospect.

The main contributions of the paper include:

(1) UbiTrack introduces a new two-way CSI method to measure distance between nodes by leveraging information from all 13 available WiFi channels, and a new data sanitization method to minimize the offset caused by hardware and the environment. UbiTrack achieved an average ranging error of 1.32 m LOS and 1.68 m NLOS using the 2.4 GHz WiFi band.

(2) UbiTrack introduces HiBay, a probabilistic approach based on Bayesian estimation to localize the target, utilizing historical position to enhance the Bayesian approach. This approach minimizes the error in distance measurement, and outperforms the traditional trilateration method. The historical location also benefits the quasi-static devices and increases their localization accuracy of 25% over time with periodic network update, having a median localization error of 1.71 m LOS and 1.93 m NLOS.

(3) UbiTrack proposes a new association-less communication method for WiFi-based localization to reduce the communication cost, this method leverages the 802.11 action frames and removes the need of association process before sending localization request to other nodes. This specifically benefits IoT devices which must minimize their active time.

2 RELATED WORK

In this section, we briefly discuss different WiFi-based indoor localization techniques that have been used to localize users and devices.

2.1 CSI-Based Indoor Localization

Many indoor localization techniques have been developed on top of commodity WiFi infrastructure. WiFi channel state information (CSI) has been used as a useful parameter to extract information about an object's location. CSI includes amplitude and phase responses of channels over different frequency providing more robust information of a wireless signal.

SpotFi [16] estimates the AoA and ToF of a target's signal using the CSI information provided by the WiFi APs and estimates the likelihood for each multipath signal to narrow down the location of the target. The solution requires a minimum of three antennas to achieve super-resolution which can be a constraint for IoT devices with single antenna. MonoLoco [26] achieves decimeter level accuracy using a single receiver by using multipath triangulation technique. The system uses the AoA, AoD and rToF parameters calculated from the multipath signals of the target and combines them with the AoA and AoD of the LoS signal to calaculate the location and orientation of the target. Chronos [29] emulates an ultra-wideband radio by transmitting multiple packets over multiple frequencies and achieves high resolution ToF measurements. The proposed technique achieves a 65 cm median error in line of sight cases. However, the frequency hopping results in high energy consumption in the range of a few joules.

All the aforementioned system require powerful WiFi NIC cards which has three or more antennas. Our system however, only require using single antenna on each side to estimate distance between nodes, which significantly reduces the hardware cost.

2.2 FTM-based Indoor Localization

Recently, Wi-fi FTM [1] feature has been used to in RF time-of-flight measurement based positioning methods. Verification [11]
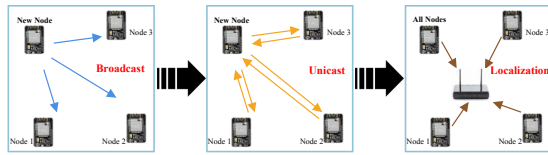
Figure 1: Overview of our proposed system. A new device transmits a broadcast message to begin a ranging event. After calculating pairwise distances, each device transmits the measurements to a nearby gateway which calculates the locations of each device.
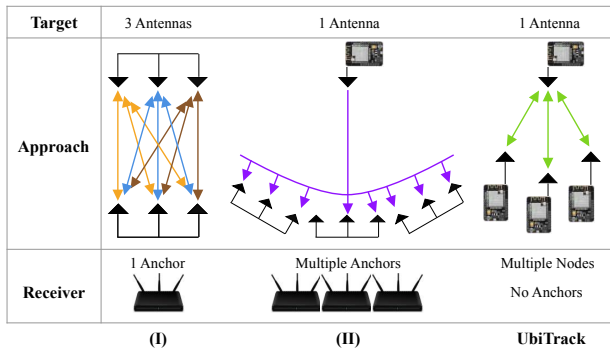

Figure 2: Comparison of WiFi-based localization approaches. Each colored line represents wireless communication between two antennas. (I) shows Chronos [29], which uses 3*3 antenna pairs. (II) shows SpotFi [16], where one node transmit packets to 3-antenna arrays on multiple anchors. Compared to the aforementioned approaches, UbiTrack only require devices to range with nearby nodes using a single antenna, being more cost and energy efficient.

confirmed that WiFi FTM based ranging can achieve up to 1m of accuracy by conducting experiments on different real world environments. The study showed that at lower bandwidth, in a NLOS settings, the ranging estimates become unreliable above distances over 20m. The paper also argued that even though commodity Wi-fi chipsets come with proper hardware support for FTM, the lack of open-source software hampers further developments using this technology. In [10], a frequency-diversity method has been introduced that can double the accuracy by leveraging the weighted averages of uncorrelated errors on different channels. The work also identified that the high relative permittivity of common building materials can introduce large distance errors making the matter worse in real environments.

3 SYSTEM OVERVIEW

UbiTrack uses a two-step process to localize all nodes in a network. First, the neighboring nodes self-organize to calculate pairwise ranges between each other. Then, all pairwise range measurements are calculated and used to estimate the relative position of each device with a newtork localization approach.

To enable pairwise ranging, UbiTrack uses a particularly useful tool of WiFi: its support for channel state information (CSI) measurements of each channel and across all subcarriers. This RF information enables better accuracy when compared to traditional

Radio Signal Strength (RSS) based approaches [2, 23, 35]. However, CSI has typically only been provided on powerful network cards like the Intel 5300 or Atheros 9580 which support multiple external antennas. Considering that most small IoT devices only cost a few dollars and use a small battery, the power consumption and hardware cost prohibits using these hardware radios on all devices in a dense IoT network. Luckily, the Espressif ESP32 [7] board is now one of the first products that enables CSI on cost-effective and energy-efficient WiFi chips.

However, small IoT devices introduce additional limitations to the deployment of the system. Small IoT devices are normally battery-powered or energy-harvesting, which limit the energy budget for communication. Thus, staying connected with neighboring devices, or going through a long WiFi association process to complete a ranging event with another device is not only energy-inefficient for a single device, but also would negatively affect the network performance for a large-scale device network. Moreover, small WiFi nodes in an IoT system are ubiquitous, and cannot always guarantee a direct line-of-sight (LOS) connection with the access point (AP), and the multipath effect in none-line-of-sight (NLOS) conditions could significantly affect the localization accuracy, thus a new method is required to overcome this limitation. What is worse, due to their form factor small devices are also unable to support a series of antennas often used in WiFi-based localization schemes [16, 26, 29, 32, 33].

UbiTrack introduces a new approach to overcome the aforementioned limitations and provides an accurate, energy-efficient localization for small IoT devices. To overcome the single antenna limitation, UbiTrack leverages CSI from all available channels in WiFi radio. UbiTrack collect CSI from both sides and apply different offset removal method to minimize the imperfections in collected CSI. To overcome the energy concern, UbiTrack introduced a new "association-less" WiFi communication method to remove the lengthy re-association process of WiFi connection to minimize packet transmission across devices. To improve the localization accuracy, UbiTrack introduces a new positioning algorithm based on historical-Bayesian estimation. Compared to traditional trilateration algorithm which suffers from non-convergence issues, our algorithm use a probabilistic model with historical priori to better localize devices.

Figure 1 shows an overview of our proposed system. When a new device joins the network, it first sends out broadcast message to neighbor nodes, where the neighbor nodes would start unicast communication with the target node. This process scans through 13 WiFi channels and measure CSI on both sides. In the next step, all devices transmit the collected CSI data to a nearby Access Point (AP), where the AP integrates all data and computes the location of the new device. Two process are required to localize a new node: ranging and positioning. The first process performs distance estimation using CSI data on both sides, afterwards the distance is sent to our proposed historical-Bayesian algorithm which estimates the position of the device. Our approach minimizes the effect of the inevitable error of distance estimation by applying a probabilistic approach with the location history of the device. A comparison of the approach of UbiTrack with similar works can be seen in Figure 2.
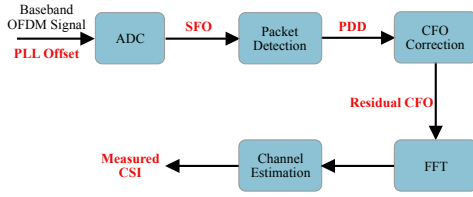
Figure 3: Different offsets in CSI. We use the zero subcarrier to minimize the impact of SFO and PDD, and measure two-way CSI to combat the offsets from CFO and PLL.

4 RANGING

In this section, we describe the procedure of measuring distance between nodes from two-way CSI measurements, and explain different methods to minimize the influence of different offsets that are causing significant errors in ranging.

4.1 Measure Distance from CSI Data

WiFi is a narrow-band wireless protocol that typically struggles to accurately measure the time-of-flight (ToF) between nodes due to its limited bandwidth (normally 20 MHz) which directly limits its resolution. As a result, traditional WiFi approaches that directly measure ToF often fail to achieve accuracy better than 50 ns (or a 15-meter distance error) [24]. Pure ToF measurement also requires tight time synchronization between nodes, which is intractable in a low-cost large-scale network. Instead, we measure the distance between nodes based on the phase shift between received WiFi signals. As a signal propagates from a transmitter to receiver, a phase shift will occur on the frequency and the distance between transmitter and receiver. A mathematical expression of the phase shift θ on a particular channel i can be written as:

$$\theta_i = 2\pi \frac{d f_i}{c} \quad mod \, 2\pi \tag{1}$$

where, d is the distance between transmitter and receiver, f_i represents the center frequency of the ith channel, and c is the speed of light. We can then extract the distance d from the above equation:

$$d = \frac{\theta c}{2\pi f_i} \pm n \frac{c}{f_i} \tag{2}$$

where, n is an integer value. Here we note that for each channel i, there can be many possible distances, each separated by a step size, c/f_i. The real distance d is the common value among possible results from all channels. A mechanism for calculating the real distance is illustrated in [30]. Mathematically, the method for finding the real distance is a form of the well-known Chinese Remainder Theorem (CRT), which can be solved using standard modular arithmetic algorithms [29]. However, we note that even after we apply the sanitization algorithm to the collected phase shifts to minimize offsets, the small error in the phase shift can still create a large error in the calculated distance. Instead of directly calculating the distance using all channels [29], we solve the ambiguity problem of CRT using remainders of coprime divisors. The IEEE 802.11 standard specifies 14 channels for WiFi communication in the ISM band, where center frequencies among channels 1-13 are separated 5 MHz apart from each other. We observe that the frequencies among every two adjacent channels are coprime, therefore, we can calculate a distance based on two neighboring channels. If there is a total of K channels that we use for distance estimation, we then compute $K-1$

measured distances, which we denote as $\hat{d_m}$ where $m \in \{1, K-1\}$. A final estimation result can be obtained by calculating the value that can achieve the minimum mean square error (MMSE):

$$d = \arg \min_x \sum_K (\hat{d_m} - x)^2 \tag{3}$$

It is also worth noting that the corresponding estimation range of our coprime pair in the 2.4 GHz band is 144 m. Therefore, it is sufficient to use in indoor localization where the ranges between devices in an IoT network are typically within tens of meters.

4.2 CSI Calibration

So far, we have explained how UbiTrack calculates distance based on CSI measurements from multiple channels. However, obtaining accurate CSI data is a challenge as the collected CSI measurement does not only describe the channel properties in passband but also the properties of the signal processing circuits in baseband. Previous studies have pointed out several major sources of CSI measurement errors that are a result of the hardware imperfection [29, 30]. Since our work only leverages the collected phase from the whole measurement, we only target errors that could create phase offsets. This includes the sampling frequency offset (SFO), packet detection delay (PDD), carrier frequency offset (CFO), and random initial phase offset. Figure 3 shows an overview of the process of CSI measurement, and explains where each error occurs along the process. We now introduce how we minimize the offset from these errors.

4.2.1 Minimizing PDD and SFO. Packet detection delay is the time between when a signal arrives and when it reaches the detection threshold. After down-converting and ADC sampling, the packet detection delay occurs due to energy detection and correlation detection in digital processing. As a result, PDD leads to a time delay in each received packet, causing CSI phase rotation error that is proportional to subcarrier index.

Sampling frequency offset is created by a difference in frequencies used in Digital to Analog Converter (DAC) and Analog to Digital Converter (ADC). This adds an extra phase shift proportional to subcarrier index and cumulative in time. The measured CSI with PDD and SFO can be expressed as:

$$\phi_{j,k} = \theta_{j,k} + k\lambda_0 + k\lambda_b \tag{4}$$

where λ_0 is the phase shift introduced by SFO, and λ_b is the phase shift introduced by PDD. Thus, if we can extract the zero subcarrier from the acquired CSI data, we can minimize the offset from this linear time delay. However, CSI measurements cannot measure the zeroth subcarrier as no data is being transmitted. Fortunately, due to the physics of signal propagation, phases are continuous over a small number of OFDM subcarriers [12]. Thus, we can use the phase in other subcarriers to approximate the phase in zero subcarrier. We applied the cubic spline interpolation method in Chronos [29] to estimate the zero subcarrier.

4.2.2 Minimizing CFO and Initial Phase Offset. Due to hardware imperfections, the carrier frequency of the transmitter and the receiver are not tightly synchronized. Although there is a CFO corrector to minimize the frequency difference, this compensation is not perfect due to hardware imperfections. Signals across all bands still carry a residual error β, which is the frequency difference between the transmitter and receiver. Besides CFO, the received CSI
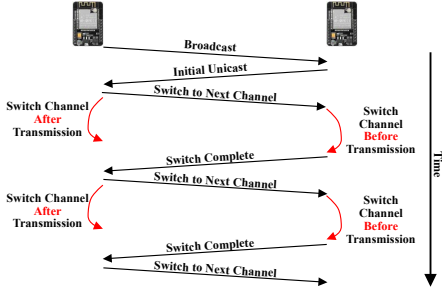
Figure 4: Association-less communication design which discards the WiFi re-association process. This method only requires two packets on each channel to measure CSI, minimizing the number of packets for ranging.

also has another additional offset, the initial phase offset. This offset is often referred as Phase-Locked Loop (PLL) phase offset, which is created during the PLL circuit generating the carrier frequency. We denote the initial phase difference between transmitter and receiver as γ. We can express the measured CSI phase of zero-subcarrier on channel m on a receiver as:

$$\hat{\theta}_{ij,m} = -2\pi f_m \tau + 2\pi\beta\tau + \gamma \tag{5}$$

Where i is the transmitter and j is the receiver. Similarly, if we have the receiver transmit back a packet, the CSI collected on the transmitter side would be:

$$\hat{\theta}_{ji,m} = -2\pi f_m \tau - 2\pi\beta\tau - \gamma \tag{6}$$

Therefore, if we add the two measured CSI and subtract by two, we can remove the offset from the CFO and the initial phase offset. However, in practice, we note that this error cannot be fully removed, as the measured CSI suffers from additional phase rotations coming from the hardware imperfections, and thus affect the accuracy on shorter distances where the offset is more significant.

4.2.3 Alleviate Multipath Effect. The wireless signals received by the receiver are not just from the direct path from the transmitter, but instead are the combination of different paths due to reflections from wall, furniture, or other objects in the space. In a multipath-weak and direct line-of-sight environment, the direct path is dominant and thus the multipath effect is minimized. However, in indoor environments, especially in residential buildings, the multipath effect can substantially affect the accuracy. We identify that the center frequencies of the 13 channels we choose have the same distance of 12 MHz between each set of two neighbors. Therefore, since the received CSI is a result of the fast Fourier transform, we can apply an inverse Fourier transform to separate the paths.

4.3 Association-less Communication

We designed a new communication method to overcome the repeating re-association process of WiFi protocol with other nodes. We leverage the vendor specific information element inside of 802.11 action frames to transmit channel switching command and data between devices. We design our two-way CSI communication in Figure 4 to minimize the extra communication process. In this process, when a device initiates the ranging request a broadcast message. Then, a number of nearby devices receive the request and respond with a unicast message specifically for the requesting device. The new node then selects three nearby nodes with highest RSSI, and

starts sending unicast message to collect CSI data with the selected nodes. Each device sends one packet on each channel to the other device. This packet contains channel switching information. The requesting device first sends a channel switching command, then switches to new channel. The anchor node receives the command, switches to the new channel, and then sends out the confirmation packet. The whole process scans through 13 available WiFi channels in the 2.4GHz band. As a fail-safe, in case where the packet is not successfully delivered to another device, we design a resend mechanism that resends the packet several times following a certain delay. If the packet is still not delivered and the receiving device has not received any packets for a certain timeout, all devices switch back to channel 1 and restart communication.

5 LOCALIZATION

To solve for the locations of all of the target nodes, we treat the coordinates of a target nodes as random variables. Localizing a target node is then equivalent to finding an assignment of coordinates that maximizes the log-likelihood in Equation (10). Directly using a convex optimization method to solve the maximization is impractical because the objective function is highly non-convex in early stages, and will be trapped into a local minimum leading to poor performance [15]. Rather than using an optimization-based approach, we use Bayesian posterior estimation to maximize Equation (10). However, conventional Bayesian estimation normally drops the prior term because of insufficient knowledge of target nodes. In doing this the system would suffer from redundant calculation for partial movement of target nodes. To improve the efficiency of Bayesian estimation and make the most of periodic updates of our system, historical knowledge is incorporated into localization. Thus, we propose a Historical Bayesian (HiBay) approach to localize the target nodes for regular system updates.

5.1 Ranges to Localization

To calculate the position of each device, we consider the network as a 2D network consisting of $n + m$ nodes with positions $c_i \in \mathbb{R}^2$, where, n are anchor nodes with known locations, m are target nodes with unknown locations, and c_i represents the coordinates of the node i in the network. As a matter of convenience, $i = 1$ to $i = m$ are targets nodes and $i = m + 1$ to $i = m + n$ are anchor nodes. The distance between nodes i and j is d_{ij}. This distance can be calculated by two nodes' locations $d_{ij} = \|c_i - c_j\|$. The observed distance o_{ij} between nodes i, j from ranging procedure is represented as

$$o_{ij} = d_{ij} + \epsilon_{ij} \tag{7}$$

where, ϵ_{ij} is the error of distance measurement which follows the Gaussian distribution. This is commonly used in modeling localization problem [27]. Using trilateration, one target node can be localized using the distances to three different anchor nodes [21]. However, there is always an error lying in distance measurement [36]. Some optimization approaches, e.g., Minimal Mean Square Error (MMSE) and Minimal Mean Absolute Error (MMAE) [4, 34], can minimize the error of localization. However, the lack of initial information of target nodes leads to poor performance of these algorithms. Instead, we estimate each node's position by a probabilistic method. The likelihood of the observed distance between two nodes i, j is given by

$$p_{ij}(o_{ij}|c_i, c_j) = \frac{1}{\sqrt{2\pi\sigma_{ij}^2}} \exp\left\{\frac{-1}{2\sigma_{ij}^2}(o_{ij} - \|c_i - c_j\|)^2\right\}, \quad (8)$$

where, σ_{ij}^2 is the variance of the observed distance o_{ij}. The coordinates of target nodes are set as random variables, so they can be estimated by maximizing the log-likelihood function of all observed distances. We define the vector O, and the coordinates set C that contains all observed distances o_{ij}, and all coordinates c_i, respectively. The joint likelihood $p(O|C)$ can be represented as

$$p(O|C) = \prod_{o_{ij} \in O} p_{ij}(o_{ij}|c_i, c_j). \quad (9)$$

The log-likelihood $l(O|C)$ of all observed distances is given by

$$l(O|C) = \ln p(O|C) = \sum_{o_{ij} \in O} \ln p(o_{ij}|c_i, c_j)$$
$$= -\sum_{o_{ij} \in O} \frac{1}{\sigma_{ij}^2}(o_{ij} - \|c_i - c_j\|)^2. \quad (10)$$

Equation (10) computes given a possible set of coordinates, how likely are the observed distances to show up. That is, a set of coordinates that maximize the possibility of observed distances are the most possible coordinates. In this way, localizing target nodes is equivalent to maximizing the Equation (10). We proposed a Bayesian approach with historical prior knowledge to solve the maximization in Section 5.

5.2 Bayesian Estimation

From a Bayesian point of view, the knowledge of unknown target nodes with coordinates C can be encoded as a probability distribution $p(C)$. Initially, we have little knowledge about the positions of target nodes. The probability distribution prior to observation, or prior distribution, is a uniform distribution meaning every target node can be anywhere within the map. After computing pairwise ranges, the knowledge about their positions changes, resulting in the posterior distribution $p(C|O)$, where O is the set of observed distances. In Bayesian estimation, posterior distribution is proportional to prior distribution multiplied by the likelihood.

$$p(C|O) \propto p(C)p(O|C), \quad (11)$$

where, $p(O|C)$ is the likelihood as described in Section 5.1. Due to the independence of different nodes, the joint prior distribution $p(C)$ and the joint posterior distribution $p(C|O)$ of all nodes' coordinates can be factorized as

$$p(C) = \prod_{i=1}^{m} p(c_i), \quad (12)$$

and

$$p(C|O) \propto \prod_{i=1}^{m} p(c_i) \prod_{o_{ij} \in O} p(o_{ij}|c_i, c_j), \quad (13)$$

respectively. Traditional Bayesian approach drops the prior term because of missing initial positions of target nodes. However, this leads to repeated calculation if the target node is slightly moved. We model the posterior distribution in a factor graph as well as incorporate historical distribution into the prior term. In this way, the posterior probability is calculated by factorizing marginal probability in the factor graph.
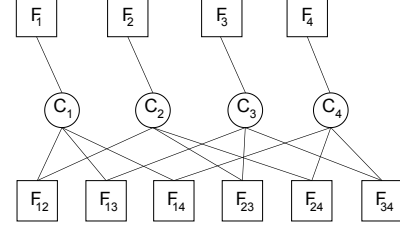

Figure 5: A factor graph example of a four-node network.

5.3 Factor Graph

A factor graph is convenient for representing the factorization of a joint probability distribution by utilizing the conditional independence of variables [19]. The network topology of the localization problem is naturally mapped to the nodes and links of the factor graph. Figure 5 shows an example of the relationship between a network consisting of four nodes and the corresponding factor graph. There are two types of nodes in the factor graph, variable nodes and factor nodes. c_i is the variable node, and F_i and F_{ij} are the factor nodes. The factor node F_i represents the prior distribution function to the variable node c_i, while the factor node F_{ij} represents the likelihood function to both variable node c_i and c_j. In this way, we can model Equation (13) as the factor graph. In the factor graph, the joint posterior distribution is given by

$$p(C|O) \propto \prod_{i=1}^{m} F_i(c_i) \prod_{(i,j) \in E} F_{ij}(c_i, c_j), \quad (14)$$

where E is the set of index pairs (i, j) that contains every pair of variable nodes connected by the same factor node. The factor node F_i and F_{ij} are defined as

$$F_i(c_i) \propto p(c_i), \quad (15)$$

and

$$F_{ij}(c_i, c_j) \propto p_{ij}(o_{ij}|c_i, c_j), \quad (16)$$

respectively. For the convenience of writing, $\mathcal{F}^k$ is used to represent the factor node of either F_i or F_{ij}, where k is the index of the factor node. With the factor graph, the posterior distribution of a variable node can be obtained by the message passing algorithm. There are two types of messages exchanged in each iteration t. $\mu_{\mathcal{F}^k \to c_i}^t$ is the message from the factor node $\mathcal{F}^k$ to the variable node c_i. $\mu_{c_i \to \mathcal{F}^k}^t$ is the message from the variable node c_i to the factor node $\mathcal{F}^k$. These are computed as follows

$$\begin{cases} \mu_{\mathcal{F}^k \to c_i}^t(c_i) = \mathcal{F}^k(c_i) \prod_{c_j \in N(\mathcal{F}^k)\backslash\{c_i\}} \mu_{c_j \to \mathcal{F}^k}^t(c_j), \\ \mu_{c_i \to \mathcal{F}^k}^t(c_i) = \prod_{\mathcal{F}^l \in N(c_i)\backslash\{\mathcal{F}^k\}} \mu_{\mathcal{F}^l \to c_i}^t(c_i), \end{cases} \quad (17)$$

where $N(c_i)$ is the set of neighbor nodes of c_i, and $\backslash\{\mathcal{F}^k\}$ is the node set excluding $\mathcal{F}^k$. The posterior distribution, or belief BL_i^t, of the variable node c_i can be obtained by multiplying all the incoming messages from the factor nodes it is connected to. The definition of BL_i^t is given as follow:

$$BL_i^t(c_i) = \prod_{\mathcal{F} \in N(ci)} \mu_{\mathcal{F} \to c_i}^t(c_i) \quad (18)$$

5.4 Historical Bayesian (HiBay) Algorithm

The conventional Bayesian localization always simplifies the prior distribution as the uniform distribution. However, there will be new adding nodes and periodic network-wide update of the localization. The Bayesian estimation will calculate the probability of positions that are far from the real position even if partial objects are slightly shifted. In order to reduce the redundant calculations, we incorporate the historical probability distribution of target nodes' positions into Bayesian estimation. It assigns low weights to the positions that are far from the previous calculated positions. The factor graph with Message Passing algorithm can naturally solve the problem. We define the algorithm is convergent at iteration t iff

$$\left\| BL_i^t - BL_i^{t-1} \right\| < \epsilon, \text{for } i = 1, 2, ..., m, \qquad (19)$$

where ϵ is the threshold of variation between the current and previous belief, and BL_i^t is the vector of beliefs over all sampling coordinates calculated by Equation (18). Suppose that, we are adding a new node or proceeding periodical system update. We have already run the HiBay algorithm for t iterations. To incorporate the historical knowledge of target positions into the current calculation of target nodes, we modify the F_i as the previous belief of node c_i

$$f_i^{upd} = BL_{c_i}^t. \qquad (20)$$

Algorithm 1 shows the our proposed HiBay algorithm. The coordinates of anchor nodes are always fixed. The factor graph can be constructed by directly mapping into the network topology. The factor node F_i, as well as the prior distribution, is at first initialized as the Uniform distribution. Then, we can draw samples of variable nodes from the factor node F_i. Starting from F_i, every node calculates and passes the messages to neighbors. Since f_i is the leaf node, it can pass the message directly to the variable node c_i without waiting for messages from other nodes. For example in Fig. 5, F_1 first passes the message $\mu_{f_1 \to c_1}$ to its neighbor c_i. Then, the factor node F_{ij} can receive the messages $\mu_{c_i \to F_{ij}}$ and $\mu_{c_j \to F_{ij}}$ from variable nodes c_i and c_j, respectively. This is because the variable node only has two neighbors. If a variable node c_i receives the message from the neighbor F_i, it can calculate and pass the message to the other neighbor F_{ij}. Finally, the factor node F_{ij} will calculate and pass the messages $\mu_{F_{ij} \to c_i}$ and $\mu_{F_{ij} \to c_j}$ to the neighbor c_i and c_j, respectively.

Above is the first iteration of message passing, while the other iterations are likewise. During the message passing process, the variable node will calculate the belief $BL_{c_i}^t$, as well as the posterior distribution of the variable c_i. Then, $\hat{c}_i^{\,t}$ can be estimated by inputting the samples to maximize the belief function. The message passing process will continue until the convergence condition is satisfied by Equation (19). If there is system-wide update or adding new node, all the factor nodes will be updated by Equation (20). Then, the message passing process is running for updated factor graph G.

6 IMPLEMENTATION

We implement a prototype of UbiTrack based on the ESP32 platform, one of the most popular WiFi IoT platforms with over 100 million devices [8], and selected the ESP-WROVER-KIT with ESP32-WROVER-E WiFi module, as shown in Figure 7. This module supports WiFi on 2.4 GHz, and is equipped with a single on-board PCB antenna. The association-less communication is implemented based

---

Algorithm 1: HiBay Algorithm

1 Initialize anchor nodes;
2 Construct factor graph G;
3 **for** *every variable node* c_i **do**
4   Draw a sample S_i of size M from f_i;
5 **end**
6 **for** *all* f_i **do**
7   Emit message $\mu_{f_i \to c_i}^1$ to neighbor node c_i;
8 **end**
9 $t := 1$;
10 **while** *not* $G.converges()$ **do**
11   **for** *each node* i *receiving messages* **do**
12   Compute messages $\mu_{i \to N(i)}^t$ to all neighbors N(i) by Eq (17)
13   **if** i *is a variable node* **then**
14   Compute belief vector $BL_{c_i}^t$ over S_i
15   by Eq (18)
16   **end**
17 **end**
18 **for** *each variable node* c_i **do**
19 $\hat{c}_i^{\,t} = \arg\max_{s \in S_i} BL_{c_i}^t(s)$
20 **end**
21 **while** *true* **do**
22   **if** *system updates or add new nodes* **then**
23   **for** *all* f_i *node* **do**
24 $G.update(f_i)$ by Eq (20)
25   **end**
26   **if** *add new nodes* **then**
27   **for** *each new node* c_j **do**
28 $G.add(c_j)$
29   **end**
30   go to $line 10$
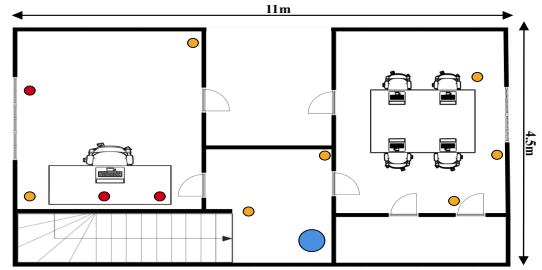**31** **end**

Figure 6: Testbed inside a resident building. Red nodes are initial anchors, yellow nodes are targets to be localized, and the blue node is the desktop for data processing.

on the ESPNOW protocol which is pre-built by Espressif [7], and leverages the vendor-specific information element in 802.11 frames to transmit information without association. We use a 100 ms delay for re-transmission for all nodes in case of a lost packet. To overcome the delay in channel switching, we design a 50 ms wait period for all receiving devices before replying with a confirmation
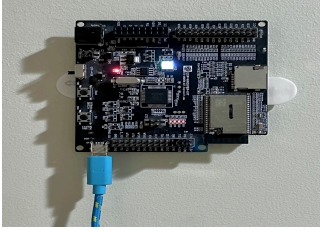
Figure 7: Node deployed on the wall. All nodes are deployed at 2 m height using an on-board PCB antenna.
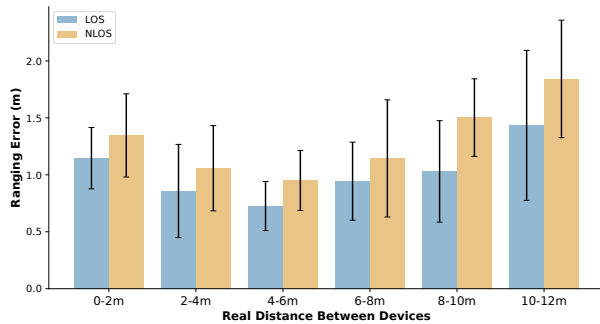

Figure 8: Ranging Accuracy. At shorter distances, remaining offsets are the major source of error. At longer distances, the propagation environment has a larger affect on accuracy.
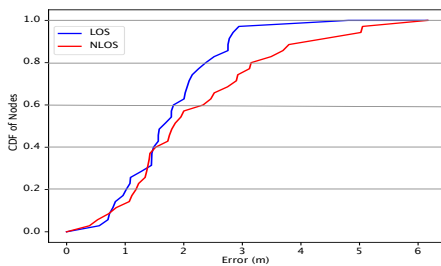

Figure 9: CDF for localization error in both LOS and NLOS scenarios. A median error of LOS is 1.71 m and the median error of NLOS is 1.93 m.

message. Devices in the network update their locations periodically, where they recalculate the distance with nearby devices and transmit the updated measurement to the central desktop. Since new devices can be added into the network, the network topology might change and the measurement will be different. We use a Linux desktop to compute the location of each node using the new measurement and the historical location as a priori.

7 EVALUATION

7.1 Experimental Setup

We deploy 10 ESP32 boards at different positions in two rooms in a residential building, as shown in Figure 6, with four devices in one room, six devices in the second room, and two devices in the hallway. The first three initial nodes are marked in red and their location is known, all other nodes participating in the localization scheme are marked in yellow. All devices are deployed at a height of 2 m to minimize the impact from furniture. Figure 7 shows our

deployment for one device on the wall. To test the system in a real environment, we did not turn off any wireless devices around our testbed, and signals from 12 adjacent access points are present in the testbed.

We tested the ranging error in both LOS and NLOS environments, and evaluated the localization error using our positioning algorithm. We also compared how different iterations can affect the accuracy of our historical-bayesian approach, and how sensitive the system is to ranging errors in potential different environments.

7.2 Ranging Accuracy

The ranging results with average error and standard deviation is shown in Figure 8. Among the measurements, we get our best results of 0.76 m error at 4-6 meters LOS and 1.03 m error at NLOS. We also noticed that ranging error increases at shorter distance of 0-2 meters, this is mainly due to the remaining offset from hardware imperfections. As distance increases, the percentage of phase shift from distance increases, however, the error from signal propagation also increases, resulting in an error of 1.43 m at 10-12 meters at LOS and 1.92 m error at NLOS. The difference in errors between LOS and NLOS scenarios is mainly due to the multipath effect. In LOS scenarios, direct path is likely to be the strongest among all communication paths. However, in NLOS scenarios, indirect path is likely to have higher power than direct path, creating an extra residual phase shift even after the CSI calibration.

7.3 Localization Error

Figure 9 plots the CDF of localization error under both LOS and NLOS scenarios. In this figure, the error of 90% nodes is within 3 m under LOS scenarios, while the error of 80% nodes is lower than 3 m in NLOS scenarios. The median error is 1.71 m for LOS scenarios and 1.93 m for NLOS scenarios, respectively. The distance estimation error with nearby nodes increases the uncertainty of localization. Due to energy constraints, the system cannot do multiple ranging rounds to increase the localization accuracy. Even under the cost and energy restrictions, the result still demonstrates that UbiTrack can provide meter-level accuracy on a budget IoT platform.

To test the probability of our proposed HiBay algorithm, we print the sampling distribution of estimated target position, where values for LOS scenarios are shown in Figure 10, and NLOS scenarios in Figure 11. The red dot and rectangular denote the true location of the target in LOS and NLOS scenarios, respectively. Comparing Figure 10 with Figure 11, we see that our system works better under LOS scenarios than NLOS scenarios. This is because higher variance of ranging error in NLOS scenarios increases the uncertainty of probability distribution of the target node's position. It is also worth noting that from iteration 1 to iteration 10, The sampling distribution gradually concentrates around the true location of the target, and the variance decreases with more iterations. The distribution stays mostly steady after 10 iterations, indicating that factor graph of the algorithm is convergent. This is where our HiBay algorithm stands out from traditional trilateration approach, the accuracy of the first iteration is similar to trilateration approaches, as there are no prior knowledge of the location. For a static device, in the following iterations, the positioning accuracy continues to increase with HiBay algorithm with the periodic update, thus resulting in a better accuracy compared to trilateration methods.
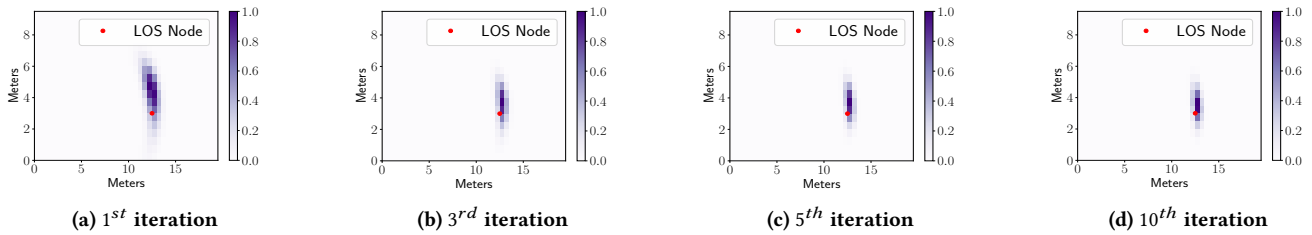
(a) 1st iteration   (b) 3rd iteration   (c) 5th iteration   (d) 10th iteration

Figure 10: Sampling distribution of estimated target position compared to true location for LOS.

(a) 1st iteration   (b) 3rd iteration   (c) 5th iteration   (d) 10th iteration
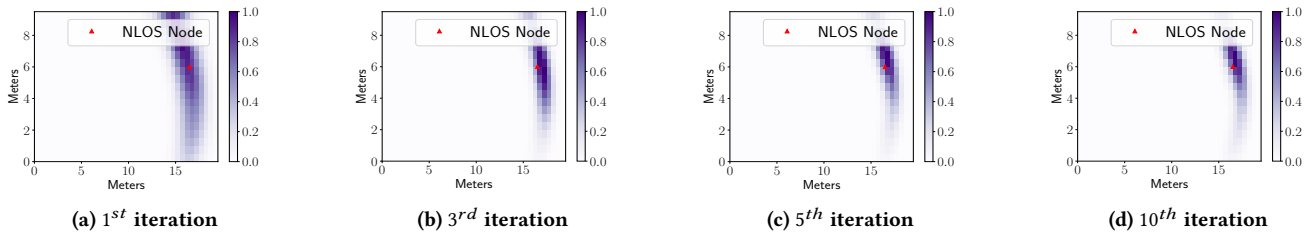
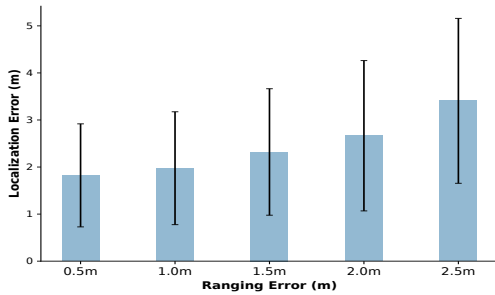**Figure 11: Sampling distribution of estimated target position compared to true location for NLOS.**

Figure 12: Localization error in response to ranging error. The results indicates that the system is stable to small variances in ranging error.
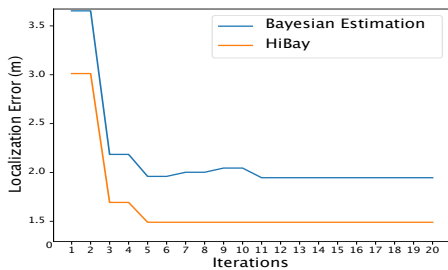

Figure 13: Performance comparison between HiBay and traditional Bayesian. HiBay achieves a quicker convergence and 25% better accuracy over traditional method.

7.4 Localization Sensitivity to Ranging Error

To test the sensitivity of localization module in response to ranging error, we simulate input distance estimation error from 0.5 m to 2.5 m, and compare the localization error in the presence of ranging errors. The results are shown in Figure 12, demonstrating that the growth of localization error is an order of magnitude slower than the increase of input ranging error. The experimental result indicates that our system is stable and robust to small errors.

7.5 Performance of HiBay compared to traditional Bayesian

Since most deployed IoT devices in buildings are static devices, we compared the performance of HiBay with traditional Bayesian estimation for static devices, and Figure 13 shows the result. Compared to traditional Bayesian estimation, HiBay converges faster, is more efficient and have achieved 25% better accuracy with the incorporation of historical data.

8 DISCUSSION

8.1 Why WiFi for Indoor Localization

Compared to BLE and UWB approaches, the WiFi approach used in UbiTrack balances cost and accuracy. Although BLE is also a prevalent wireless technology, it does not provide the fine-grained CSI information and performs poorly in complex multipath environments. Many BLE based approaches use Received Signal Strength (RSS) [14, 18], which is influenced by shadowing effects due to the presence of obstacles and reflections of typical indoor environments. UWB is another wireless technology used in indoor localization. UWB devices transmit nanoseconds short-duration pulses that can be used to provide accurate localization. Currently, the cost of UWB is much more than WiFi [5, 9], making it more expensive to deploy. There are some approaches that combine WiFi with UWB to balance cost and accuracy [20], which could be a potential solution for future indoor localization.

8.2 Energy Consumption

Energy consumption is a key factor that limits the use of some indoor localization approaches on small IoT devices. It is difficult to provide a fair energy comparison across different approaches with different wireless protocol and operational techniques. Instead, we compare the current of both TX and RX on WiFi and UWB. Currently, the ESP32 WiFi chip consumes an average of 190 mA of current in TX and 98 mA in RX [6], while the DW1000 UWB chip consumes an average of 140 mA of current in TX and 193 mA in RX [28]. The WiFi approach requires more packet transmission, leveraging multiple channels to achieve a closer accuracy compared

to UWB approach. There are works that aims to tackle this problem by reducing the energy consumption of WiFi chips and by introducing new protocols to achieve higher accuracy with fewer transmissions. Adopting low-power hardware could save both transmission and standby power by more than 20% [31]. Additionally, new WiFi protocols like FTM [1] have great potential to provide sub-meter level ranging with possibly less energy consumption.

8.3 Limitations and Future Work

One limitation of UbiTrack is the use of the on-board PCB antenna on ESP32 chips. As PCB trace antennas are highly susceptible to environmental noise [13], during experiments we found that transmission often required frequent re-transmissions during a ranging measurement. We will explore how PCB antennas can affect the ranging accuracy compared to external antennas used on other approaches.

In our next step, we plan to look into more opportunities that can increase the localization accuracy and reduce the number of transmissions of WiFi-based approaches. One direction is to see if combining CSI with FTM together can achieve a better performance compared to use these techniques separately. Another direction is to explore energy efficient protocols for WiFi ranging. For example, adding a wake-up receiver can significantly reduce the standby energy of WiFi with sacrificing a slight overhead at wake-up [31].

9 CONCLUSIONS

UbiTrack provides a scalable & low-cost network localization service for devices using onboard wifi. Compared to other approaches using WiFi NIC cards using array of antennas, UbiTrack only requires one single antenna on both devices, and is more cost-effective and energy-efficient. UbiTrack leverages the already densely deployed IoT devices and proposes HiBay, a historical-bayesian network localization algorithm that helps to increase the overall localization accuracy. UbiTrack has demonstrated meter-level accuracy in indoor environments, and opens up great potential for location-based services which are often not included in today's IoT devices.

10 ACKNOWLEDGEMENT

REFERENCES

[1] IEEE standard for information technology—telecommunications and information exchange between systems local and metropolitan area networks—specific requirements - part 11: Wireless lan medium access control (mac) and physical layer (phy) specifications. *IEEE Std 802.11-2016 (Revision of IEEE Std 802.11-2012)*, pages 1–3534, 2016.

[2] P. Barsocchi, S. Lenzi, S. Chessa, and G. Giunta. A novel approach to indoor rssi localization by automatic calibration of the wireless propagation model. In *VTC Spring 2009-IEEE 69th Vehicular Technology Conference*, pages 1–5. IEEE, 2009.

[3] X. Cai, L. Ye, and Q. Zhang. Ensemble learning particle swarm optimization for real-time uwb indoor localization. *EURASIP Journal on Wireless Communications and Networking*, 2018(1):125, 2018.

[4] W. Cui, L. Zhang, B. Li, J. Guo, W. Meng, H. Wang, and L. Xie. Received signal strength based indoor positioning using a random vector functional link network. *IEEE Transactions on Industrial Informatics*, 14(5):1846–1855, 2017.

[5] Decawave. DW1000 chip $10.92. https://www.digikey.com/, 2021.

[6] Espressif. ESP32 Datasheet. https://www.espressif.com/sites/default/files/documentation/esp32_datasheet_en.pdf.

[7] Espressif. ESP32 MCU. https://www.espressif.com/en/products/socs/esp32.

[8] Espressif. Espressif Achieves 100-Million IoT Chip Target. https://www.espressif.com/en/news, 2018.

[9] Espressif. Esp32-S2 chip $1.35. https://www.digikey.com/, 2021.

[10] B. K. P. Horn. Doubling the accuracy of indoor location: Frequency diversity. 2020.

[11] M. Ibrahim, H. Liu, M. Jawahar, V. Nguyen, M. Gruteser, R. Howard, B. Yu, and F. Bai. Verification: Accuracy evaluation of wifi fine time measurements on an open platform. In *Proceedings of the 24th Annual International Conference on Mobile Computing and Networking*, pages 417–427, 2018.

[12] IEEE. 802.11n Standard. http://standards.ieee.org/findstds/standard/802.11n-2009.html, 2009.

[13] T. Instrument. PCB Design Guidelines For Reduced EMI. https://www.ti.com/lit/an/szza009/szza009.pdf, 1999.

[14] Z. Jianyong, L. Haiyong, C. Zili, and L. Zhaohui. Rssi based bluetooth low energy indoor positioning. In *2014 International Conference on Indoor Positioning and Indoor Navigation (IPIN)*, pages 526–533. IEEE, 2014.

[15] R. Koenker and I. Mizera. Convex optimization, shape constraints, compound decisions, and empirical bayes rules. *Journal of the American Statistical Association*, 109(506):674–685, 2014.

[16] M. Kotaru, K. Joshi, D. Bharadia, and S. Katti. Spotfi: Decimeter level localization using wifi. In *Proceedings of the 2015 ACM Conference on Special Interest Group on Data Communication*, pages 269–282, 2015.

[17] S. Krishnan, P. Sharma, Z. Guoping, and O. H. Woon. A uwb based localization system for indoor robot navigation. In *2007 IEEE International Conference on Ultra-Wideband*, pages 77–82. IEEE, 2007.

[18] Z. Ma, S. Poslad, J. Bigham, X. Zhang, and L. Men. A ble rssi ranking based indoor positioning system for generic smartphones. In *2017 Wireless Telecommunications Symposium (WTS)*, pages 1–8. IEEE, 2017.

[19] Y. Mao, F. R. Kschischang, B. Li, and S. Pasupathy. A factor graph approach to link loss monitoring in wireless sensor networks. *IEEE Journal on Selected Areas in Communications*, 23(4):820–829, 2005.

[20] S. Monica and F. Bergenti. Hybrid indoor localization using wifi and uwb technologies. *Electronics*, 8(3):334, 2019.

[21] O. Oguejiofor, A. Aniedu, H. Ejiofor, and A. Okolibe. Trilateration based localization algorithm for wireless sensor network. *International Journal of Science and Modern Engineering (IJISME)*, 1(10):2319–6386, 2013.

[22] K. Qian, C. Wu, Z. Yang, C. Yang, and Y. Liu. Decimeter level passive tracking with wifi. In *Proceedings of the 3rd Workshop on Hot Topics in Wireless*, pages 44–48, 2016.

[23] S. Sadowski and P. Spachos. Rssi-based indoor localization with the internet of things. *IEEE Access*, 6:30149–30161, 2018.

[24] S. Sen, J. Lee, K.-H. Kim, and P. Congdon. Avoiding multipath to revive inbuilding wifi localization. In *Proceeding of the 11th annual international conference on Mobile systems, applications, and services*, pages 249–262, 2013.

[25] B. SIG. Bluetooth Standard. https://www.bluetooth.com/.

[26] E. Soltanaghaei, A. Kalyanaraman, and K. Whitehouse. Multipath triangulation: Decimeter-level wifi localization and orientation with a single unaided receiver. In *Proceedings of the 16th annual international conference on mobile systems, applications, and services*, pages 376–388, 2018.

[27] S. Thrun. Simultaneous localization and mapping. In *Robotics and cognitive approaches to spatial mapping*, pages 13–41. Springer, 2007.

[28] B. Van Herbruggen, B. Jooris, J. Rossey, M. Ridolfi, N. Macoir, Q. Van den Brande, S. Lemey, and E. De Poorter. Wi-pos: A low-cost, open source ultra-wideband (uwb) hardware platform with long range sub-ghz backbone. *Sensors*, 19(7):1548, 2019.

[29] D. Vasisht, S. Kumar, and D. Katabi. Decimeter-level localization with a single wifi access point. In *13th {USENIX} Symposium on Networked Systems Design and Implementation ({NSDI} 16)*, pages 165–178, 2016.

[30] C. Wang, Q. Yin, and H. Chen. Robust chinese remainder theorem ranging method based on dual-frequency measurements. *IEEE Transactions on Vehicular Technology*, 60(8):4094–4099, 2011.

[31] W. Wang, J. Su, Z. Hicks, and B. Campbell. The standby energy of smart devices: Problems, progress, & potential. In *2020 IEEE/ACM Fifth International Conference on Internet-of-Things Design and Implementation (IoTDI)*, pages 164–175. IEEE, 2020.

[32] X. Wang, L. Gao, S. Mao, and S. Pandey. Csi-based fingerprinting for indoor localization: A deep learning approach. *IEEE Transactions on Vehicular Technology*, 66(1):763–776, 2016.

[33] K. Wu, J. Xiao, Y. Yi, D. Chen, X. Luo, and L. M. Ni. Csi-based indoor localization. *IEEE Transactions on Parallel and Distributed Systems*, 24(7):1300–1309, 2012.

[34] L. Xiao, L. J. Greenstein, and N. B. Mandayam. Sensor-assisted localization in cellular systems. *IEEE Transactions on Wireless Communications*, 6(12):4244–4248, 2007.

[35] S. Yiu, M. Dashti, H. Claussen, and F. Perez-Cruz. Wireless rssi fingerprinting localization. *Signal Processing*, 131:235–244, 2017.

[36] L. Zhu, Z. Xiong, H. Ding, and Y. Xiong. A distance function based approach for localization and profile error evaluation of complex surface. *J. Manuf. Sci. Eng.*, 126(3):542–554, 2004.