SPRING+: Smartphone Positioning From a Single WiFi Access Point

Stavros Eleftherakis, Giuseppe Santaromita, *Member, IEEE*, Maurizio Rea, Xavier Costa-Pérez, *Senior Member, IEEE*, and Domenico Giustiniano, *Senior Member, IEEE*

***Abstract*—Indoor positioning is a major challenge for location-based services. WiFi deployments are often used to address indoor positioning. Yet, they require *multiple access points*, which may not be available or accessible for localization in all scenarios, or they make unrealistic assumptions for practical deployments. In this paper we present *SPRING+*, a positioning system that extracts and processes Channel State Information (CSI) and Fine Time Measurements (FTM) from a single Access Point (AP) to localize commercial smartphones. First, we propose an adaptive method for estimating the Angle of Arrival (AOA) from CSI that works on single packets and leverages information from the estimated number of paths. Second, we present a new method to detect the first path using FTM measurements, robust to multipath scenarios. We evaluate SPRING+ in an extensive experimental campaign consisting of four different testbeds: i) generic indoor spaces, ii) generic indoor spaces with obstacles, iii) office environments and iv) home environments. Our results show that SPRING+ is able to achieve a median 2D positioning error between 1 and 1.8 meters with a *single WiFi AP*.**

***Index Terms*—Indoor localization system, WiFi, angle of arrival, channel state information, fine time measurements, smartphone localization.**

I. Introduction

THE popularity of location-based services is exponentially increasing thanks to the industrial interest and standardization effort of positioning solutions in 5G networks. Indoor localization, however, remains a major challenge where people experience a very poor performance due to the current technology limitations. Even with the advent of 5G networks, WiFi represents the main technology solution for indoor scenarios as its measurements are expected to be accessed by 5G positioning systems [1], [2].

Accurate indoor positioning can be achieved through WiFi using different approaches based on signal strength [3], [4], [5], [6], [7], [8], Angle of Arrival (AOA) [9], [10], [11], [12], time-based ranging [13], [14], [15], [16]. Based on this vast literature, *localizing a device with a dense network of Access Points (APs) is a solved problem, yet this is not the case when a single AP is available*. Unfortunately, such a case is frequent in practice and severely limits the deployment of indoor location-based services. For instance, homes and small businesses typically have a single AP. In other scenarios, other APs could be seen in range, but their signals could be too weak for collecting positioning data.

Early attempts to operate with a single AP assumed that the WiFi chipset in the AP can continuously change its frequency of operation [17], [18], which is not supported by neither any 802.11 standard nor smartphone. [19] required usage of inertial sensors to work with a single AP, as well as extensive manual calibration, such as placing the phone in the pocket for training. Other work requires open areas, which is not practical in real deployments [20]. Furthermore, HiLoc [21] extracted ToF measurements from a prototype AP design, constructed by a USRP with six antennas connected with extension cables and cannot work in NLOS scenarios. Besides, fingerprinting methods that fuse different signal characteristics and work with one AP are vulnerable to environmental changes [22], [23]. All the above factors limit the deployment at a large scale, showing that working with a single AP continues to be one of the "Achilles heels" for indoor localization systems.

In this work, we introduce SPRING+, Smartphone Positioning with Radio measurements from a sINGle wifi access point. First, we take advantage of the 802.11ac standard, extracting Channel State Information (CSI), obtained from multiple antennas in the overall 80 MHz bandwidth. Second, recently, IEEE 802.11 standardized the FTM protocol [24] to support time-based WiFi ranging techniques. So far, there have been few studies of such ranging system, with limited investigation of methods to alleviate the multipath. Using CSI and FTM measurements, we estimate the angle and the distance from an AP to a commercial off-the-shelf smartphone. Angle and distance intersect in a single position, which provides the location of the smartphone. We are the first to show the feasibility of locating a

Manuscript received 4 August 2023; revised 29 November 2023; accepted 6 February 2024. This work has been funded at IMDEA Networks in part by the project MAP-6G, reference TSI-063000-2021-63, granted by the Ministry of Economic Affairs and Digital Transformation and the European Union-NextGenerationEU through the UNICO-5G R\&D program of the Spanish Recovery, Transformation and Resilience Plan, and in part by project PID2022-136769NB-I00 (6th-SENSE_ELSA) funded by MCIN/AEI/10.13039/501100011033/ and by the ERDF, A way of making Europe. This work has been funded at i2CAT foundation, NEC Labs Europe and ICREA in part by the UNICO I+D 5G 2021, reference number TSI-063000-2021-6 and in part by EU SNS JU Project INSTINCT. Recommended for acceptance by T. Shu. *(Corresponding author: Stavros Eleftherakis.)*

Stavros Eleftherakis is with the Telematics Engineering Department of Universidad Carlos III de Madrid (UC3M), 28911 Leganes, Madrid, Spain, and also with IMDEA Networks Institute, 28918 Leganes, Madrid, Spain (e-mail: stavros.eleftherakis@imdea.org).

Giuseppe Santaromita and Domenico Giustiniano are with the IMDEA Networks Institute, 28918 Leganes, Madrid, Spain.

Maurizio Rea is with the i2cat Foundation, 08034 Barcelona, Spain.

Xavier Costa-Pérez is with the i2cat Foundation, 08034 Barcelona, Spain, also with NEC Labs Europe, 69115 Heidelberg, Germany, and also with ICREA, 08010 Barcelona, Spain.

Digital Object Identifier 10.1109/TMC.2024.3367241

smartphone in typical indoor environments with measurements made only by the single AP, and quantify the performance that can be achieved.

In the following, we summarize our contributions:

- We introduce our positioning system SPRING+ that uses AOA, Relative (R-ToF) and Absolute Time of Flight (A-ToF) information to estimate the angle and distance, and localize the smartphone using a single AP. As multipath affects both angle and distance estimates, we use its estimate as input for both our AOA and distance estimators (Section III).
- We propose a direct AOA estimator that processes CSI measurements and dynamically selects the optimal configuration in the 2D space composed by AOA and Relative Time of Flight, addressing major limitations from prior work such as static selection of smoothing length and number of clusters in a packet (Section IV);
- We analyze the key factors and parameters that affect the ranging performance using the FTM protocol. We show how to mitigate the impact of multipath relying on inputs from CSI, developing an estimator that improves the performance in the presence of rich multipath as found in common indoor environments (Section V);
- We demonstrate across four different experimental testbeds that smartphones can be localized using our solution with a median accuracy between 1 and 1.8 meters in areas up to 143 m^2 in a single testbed, and a total of 373 m^2 of area evaluated (Sections VI and VII).

We highlight that SPRING+ is an extension of our preliminary short paper work SPRING [25], and its code will be published as an open source code in GitHub[1]

II. Related Work

Homes and small businesses typically have a single AP, and our proposed system, SPRING+, aims to provide a positioning solution that works with a single AP. There is a variety of different approaches related to positioning, and, in what follows, we briefly analyze the key contributions related to our work.

A first well-known approach is the RSSI-based systems [3], [5], [7], [26], [27], [28], [29], [30], in which the signal strength from a client device to several different APs is measured. RADAR [3] achieved a median error of around 3 meters, using three Base Stations, unlike SPRING+, and relies on extensive measurement of the radio frequency environment. Besides, [28] worked with a density of one AP per 16 m^2 for a common indoor space, whereas SPRING+ has been tested to work with a single AP in areas up to 143 m^2. Moreover, EZ [7] combined Global Positioning System (GPS) and RSSI information deploying four APs in order to achieve a median localization error of 2 meters. Apart from these, [29] needs access to measurements obtained by motion sensors (inertial sensors and magnetometers), thus setting an important limitation compared to SPPRING+ that works only with commodity hardware and does not need any type of sensor. Finally, [27] and [30] used Reconfigurable Intelligent Surfaces (RIS) to generate the radio environment and create a favorable RSS distribution. This equipment increases the cost and limits the scalability of the system. Compared to the aforementioned methods, SPRING+ is a low-cost and easily deployable solution that has been tested using commodity equipment and with only one AP, with experiments covering an indoor spaces of up to 143 m^2.

Fingerprinting methods are also very well-investigated [4], [8], [31], [32], [33], [34], [35], [36], [37], [38], [39], [40], [41], [42], [43], [44], [45]. The main idea of fingerprinting methods is the collection of fingerprints, such as RSSI or CSI, from a specific target and then any other unknown signals are compared to the predefined database of signal patterns. These methods are not easily deployable and they are also very sensitive to any environmental change. For instance, Centaur [35] combined WiFi-based (Horus [4], EZ [7]) and acoustic ranged techniques [46] to achieve a localization error of around 3.4 meters. Other systems like Zee [8] and Unloc [34], use either inertial or urban sensors. Furthermore, there are CSI fingerprinting methods that achieve a reasonable accuracy [36], [37], [40], [45]. In contrast to SPRING+, both the construction of the fingerprint database and the use of a dense network of APs limit the deployment capability of these systems and increase the cost of their implementation. Besides, RRLOC [42] fuses RSSI and FTM measurements to construct a feature database and then trains a Machine Learning model to localize the target device. The aforementioned system not only needs offline training to construct the database, but its localization error is around 4 meters with one or two APs, thus being less accurate than SPRING+. In addition, [43] proposes a manual modification of both the orientation and polarization of the AP antennas to construct a more precise RSS fingerprinting database, at the cost of system scalability. Finally, [44] proposes a DL-based CSI fingerprinting method, where robots carry the APs in the indoor environment for the database construction. As opposed to SPRING+, this method has higher deployability cost and similar to all fingerprinting methods its database is prone to environmental changes.

AOA and CSI-based solutions have been investigated [9], [11], [12], [20], [47], [48], [49], [50], [51], [52], [53], [54], [55], [56], [57], [58]. They take advantage of the CSI measurements to calculate the AOA between the target and the AP. Even if their accuracy is encouraging, most of the times these solutions are not easily deployable, as they require a dense deployment of APs. For instance, Array Track [9] and SpotFi [12] deployed a very dense number of APs, making, in contrast to SPRING+, unfeasible the deployment in small environments. Besides, the systems proposed in [53], [54] estimate AOA using UWB transceivers equipped with modified antennas or specific hardware (CISCO 4800 series) respectively, thus being less practical and more expensive than SPRING+. A recent attempt, called Ubilocate [51], works with a variety of signal characteristics and a dense network of APs. Moreover, Ubilocate, unlike our proposed method based on FTM, used a non-standard procedure to estimate the distance and its performance for AOA estimation is compared to SPRING+ in Section VII-B. Furthermore, Widar2.0 [20] is a WiFi-based system used for passive human tracking that works with only one commodity AP. However, it works only in open areas without obstructions between the AP and the target device, a requirement that SPRING+ does not have. mD-Track [52] is another recent work that focuses on passive WiFi tracking, using several dimensions (AOA, A-ToF,

[1] Online. [Available]. https://github.com/stavroseleftherakis/SPRING_PLUS

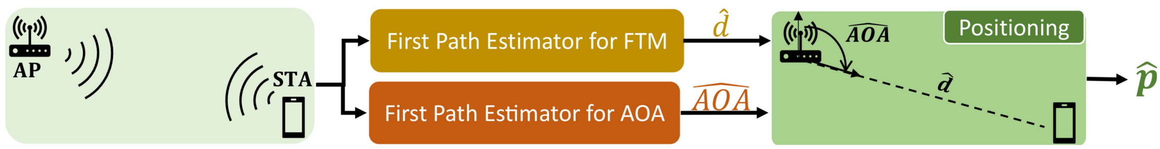


Fig. 1. Overview of SPRING+'s building blocks.

R-ToF, Angle of Departure, Doppler). However, it focused on one single passive target, and it did not have access to A-ToF real measurements, using coaxial cables for this purpose. Besides, [55] combines sub-6 GHz measurements with mmWave ones to construct an accurate localization system, but it cannot be compared with SPRING+ that works only with commodity equipment in the 5-GHz band. Apart from these, SiFi [58] is a single AP localization system, that works with a customized AP where antennas are connected to the AP through a 5 meters long extension cable, must not be collinear to each other and they also need to have a distance of 2-3 meters between them. Obviously, these are limitations that SPRING+ does not have, as our proposed system works with a commodity ULA AP. In addition, [56] is another single AP localization system, that requires either an additional WiFi card or coaxial cables connecting the transceivers' antennas only to calibrate the initial CSI phases multiple times and has been tested to work in areas up to 85 m^2. In contrast, SPRING+ needs neither additional equipment for calibration nor multiple calibration steps and has been tested to work in larger areas, thus being more easily deployable. Finally, [57] is another recent CSI localization method that requires a specific antenna layout that is not compatible with the existing WiFi APs, whereas SPRING+ work with a commodity ULA AP.

Finally, A-ToF based techniques are used to estimate the distance and position of the target device [15], [19], [59], [60], [61], [62], [63], [64]. First, [19] and [64] are A-ToF systems using CSI or FTM respectively, but both require access to inertial sensors in the smartphone, a limitation that SPRING+ does not have. Moreover, [59] combined multiple frequency bands scattered around 2.4 GHz and 5 GHz, to obtain almost one GHz of bandwidth. Yet, this approach can not work with commercial smartphones as they operate on a single frequency channel. In contrast, SPRING+ has been tested with 80 MHz bandwidth and single frequency channel, using a commercial smartphone as a target. Besides, [62] and [63] are A-ToF systems that use four UWB APs for human tracking, but require humans to wear special sensor equipment in their chests, thus limiting the deployment of the system and increasing its cost. Finally, FUSIC [61] combines an improved version of FTM and a Least Squared Optimization problem, using three APs to achieve a median localization error between 1.74 m and 2.19 m. It is not possible to compare their positioning estimator with SPRING+, since their system cannot work with less than three APs, but we show in Section VII-B that our proposed ranging estimator outperforms this prior solution.

III. SPRING+ Overview

State-of-the-art indoor positioning solutions rely on different approaches which combine measurements collected by different APs densely deployed. Other solutions operate with a single AP, either assuming particular features not supported by commodity

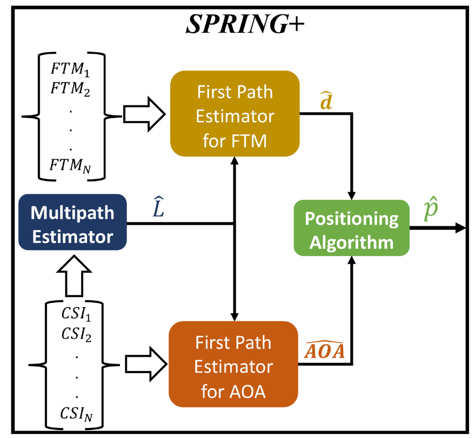


Fig. 2. Schematic of operation of SPRING+ localization system. Number of estimated paths are used both by our AOA and FTM (distance) first path estimators.

smartphones, or requiring extensive calibrations and computation that limit the deployment of these approaches at large scale. In contrast, hereafter we present SPRING+, that aims to locate commercial smartphones in typical indoor environments, only with measurements performed by a single AP. The building blocks of SPRING+ high level overview are depicted in Fig. 1.

First Path Estimator for AOA: The AP is equipped with an array of multiple antennas, and it measures CSI data from the smartphone, indicated in the figure as STAtion (STA). CSI allows us to estimate AOA and the *Relative Time of Flight (R-ToF)* for each path. With this information, we propose a first path estimator to identify the direct path signal component, as detailed in Section IV. No protocol stack modifications are needed such that compatibility with commercial phones is guaranteed, as described in Section VI-A.

First Path Estimator for FTM: The AP supports the FTM protocol standardized by IEEE 802.11mc in 2016 and described in Section V-A. This protocol allows measuring the *Absolute Time of Flight (A-ToF)* between the AP and the STA. The main difference between A-ToF and R-ToF is that A-ToF is an absolute measurement of the round trip time to transmit a series of packets between AP and smartphone, hence providing an estimate of the real distance between the AP and the smartphone. Instead, R-ToF is only a relative measurement of the ToF, used to better distinguish among different paths, but it does not carry information about the real distance. Using our A-ToF measurements, we introduce a first path estimator to mitigate the effect of the multipath, typical characteristic of rich indoor environments, as explained in detail in Section V.

Positioning: Following the SPRING+'s building blocks in Fig. 1, we finally calculate the smartphone's position. More specifically, angle and distance are estimated by SPRING+ and intersect in a single position, which provides the location of the smartphone. In Fig. 2 we see a schematic of SPRING+ operation.

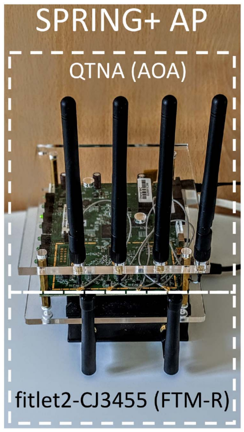


Fig. 3. SPRING+ AP used for measurements.

As multipath affects both angle and distance estimations, we use the estimated number of paths $\hat{L}$ per packet as input in *both* the AOA and FTM first path estimators. The outputs are the estimated distance ($\hat{d}$) and direction ($\widehat{AOA}$) between the target and the AP, respectively. Using the parameters $\hat{d}$ and $\widehat{AOA}$, SPRING+ then estimates the target position, $\hat{p}$.

An illustration of the commodity hardware we use for the AP is shown in Fig. 3. The AP for the experimentation is composed of two different commodity chipsets (although early attempts for an implementation in the same chipset have just started [65]). The smartphone STA is associated to the chipset performing AOA, while FTM does not require association of the STA [24]. Therefore, it is possible to use the same STA, such as a commodity smartphone, for both type of measurements as it would be done from a single WiFi chipset in the AP through normal listen-before-talk 802.11 protocol.

IV. First Path Estimator for AoA

This section presents the proposed algorithm used to identify the direct AOA. A high-level illustration of the scheme is illustrated in Fig. 4.

A. An Intuition to CSI Smoothing

The proposed AOA estimator takes as input data the CSI measurements extracted from WiFi commodity chipsets. This data is organized in a matrix that contains one complex number $\mathrm{CSI}_{i,j}$ per subcarrier and per received antenna at the AP. It is of dimensions MxN_s, where M is the number of antennas and N_s is the number of subcarriers, i.e.:

$$\begin{bmatrix} \mathrm{CSI}_{1,1} & \dots & \mathrm{CSI}_{N_s,1} \\ \vdots & \ddots & \vdots \\ \mathrm{CSI}_{1,M} & \dots & \mathrm{CSI}_{N_s,M} \end{bmatrix} \tag{1}$$

This input matrix is composed of calibrated values in phase and amplitude, where the bias due to a phase shift among the antennas for all N_s subcarriers is corrected. We use the method proposed in our preliminary work [25], that runs the procedure only once (not detailed here for the sake of brevity).

Starting from the input CSI matrix, we construct the smoothed CSI matrix $\mathbf{X}$ for each collected packet $s \in [1, N]$, where N is the total number of packets. The smoothed CSI matrix increases the resolution of input data by grouping different subsets of consecutive antennas K and subcarriers J together in each column [12]. The smoothed CSI is composed of Hankel matrices H of the input CSI matrix, wherein the elements along each anti-diagonal are equal. For instance, $\mathrm{H}_{1,J}$ takes J consecutive subcarriers of the antenna one, starting from the first subcarrier:

$$\mathrm{H}_{1,J} = \begin{bmatrix} \mathrm{CSI}_{1,1} & \mathrm{CSI}_{1,2} & \dots & \mathrm{CSI}_{1,N_s-J+1} \\ \mathrm{CSI}_{1,2} & \mathrm{CSI}_{1,3} & \dots & \mathrm{CSI}_{1,N_s-J+2} \\ \vdots & \vdots & \ddots & \vdots \\ \mathrm{CSI}_{1,J} & \mathrm{CSI}_{1,J+1} & \dots & \mathrm{CSI}_{1,N_s} \end{bmatrix} \tag{2}$$

Similarly, $\mathrm{H}_{K,J}$ takes J consecutive subcarriers of the antenna K, starting from the first subcarrier. It results that the smoothed CSI matrix can be written as follows:

$$\mathbf{X} = \begin{bmatrix} \mathrm{H}_{1,J} & \dots & \mathrm{H}_{K,J} \\ \vdots & \ddots & \vdots \\ \mathrm{H}_{K,J} & \dots & \mathrm{H}_{N_s,J} \end{bmatrix} \tag{3}$$

Having as input the CSI matrix of dimensions N_sxM, the dimensions of the new smoothed CSI matrix are equal to $(K \cdot J)$ x $(N_s + 2)$, where K and J are the parameters of the smoothing algorithm and their product is called *smoothing length* [12]. Note that the maximum achievable smoothing length is equal to $N_s \cdot M$. In the next subsection, we present our approach to optimize the smoothing matrix.

B. Optimization of Smoothing Matrix

Prior work neglected the importance of the size of different subsets of input data to group, and always fixed it to a static number. In an earlier attempt, ArrayTrack [9] applied grouping only to antennas and fixed the value to two. SpotFi [12] extended the concept of grouping to both antennas and subcarriers. Although not specified in [12], but only in the source code, SpotFi used a fixed smoothing length, with $K = (\frac{M}{2}+1)$ and $J = \frac{N_s}{2}$, thus treating a smoothed CSI matrix with a fixed dimension equals to $K \cdot J = (\frac{M}{2}+1) \cdot \frac{N_s}{2}$.

In order to understand the impact of this choice, we compute the 2D MUltiple SIgnal Classification (MUSIC) algorithm [66] using the smoothed CSI $\mathbf{X}$ as input data. MUSIC performs a subspace decomposition of the autocorrelation matrix $\mathbf{XX^H}$. It defines the spatial spectrum by multiplying a set of steering vectors by the noise subspace, orthogonal to each other. In the 2D MUSIC spectrum, the x-axis indicates the estimated R-ToF and the y-axis the estimated AOA.

We study the 2D MUSIC spectrum for two different positions of the STA. In the first one, as shown in Fig. 5(a), choosing the values K and J as set in prior work results in a higher noise level and larger beam amplitude and width of all the paths in the spectrum. The consequence is that a mistaken beam is considered as the real path, leading to an AOA error of around 43 degrees. In the second one, we instead apply the simple approach of considering the highest smoothing length, equal to

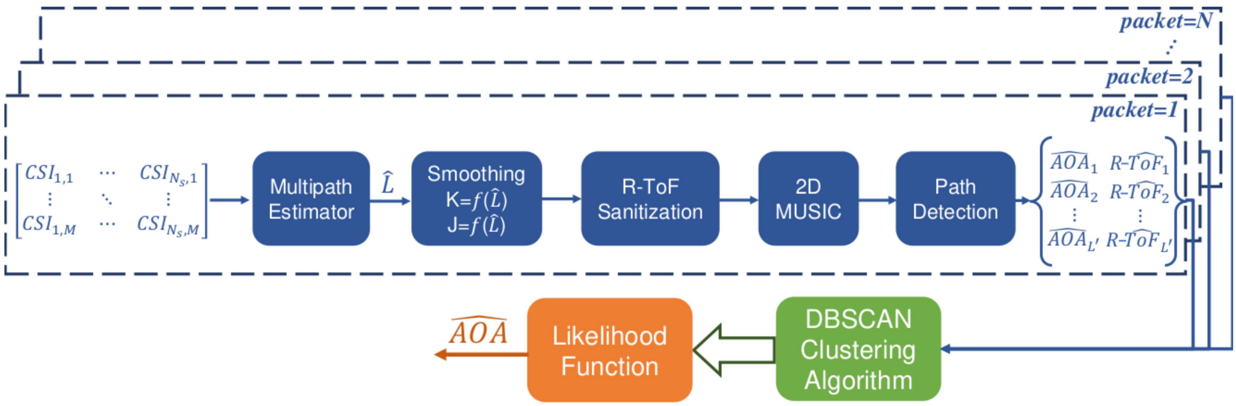
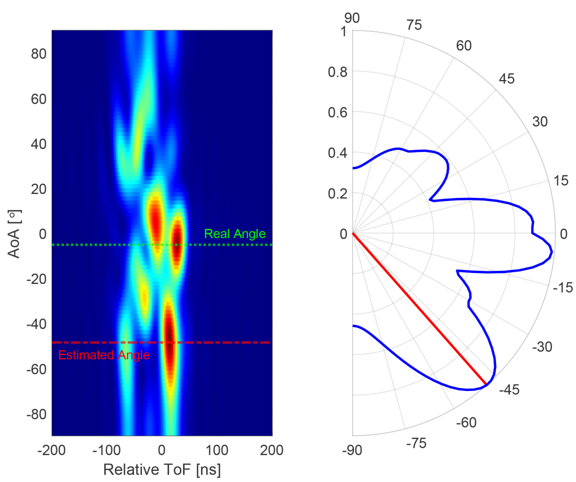
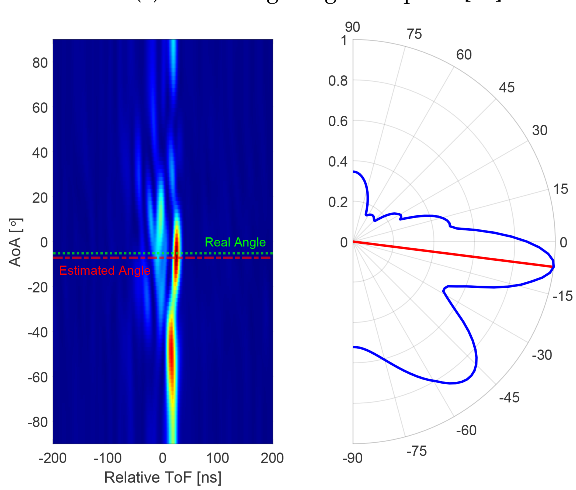


Fig. 4. Illustration of the proposed First Path AoA Estimator.

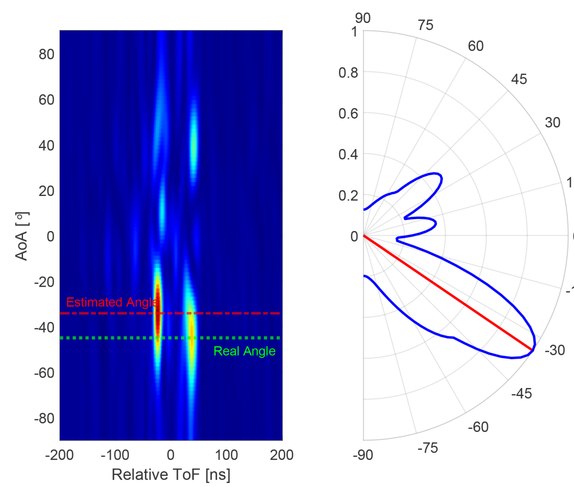


(a) Smoothing Length in SpotFi [12].

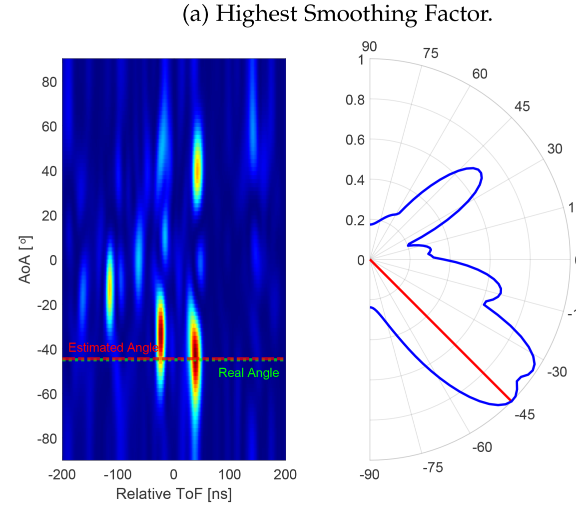


(b) Proposed multipath-based smoothing length.

Fig. 5. 2D MUSIC spectrum (on the left) and beam patterns (on the right) for two different smoothing lengths for a given position of the STA.



(a) Highest Smoothing Factor.



(b) Proposed multipath-based smoothing length.

Fig. 6. 2D MUSIC spectrum (on the left) and beam patterns (on the right) for two different smoothing lengths for a given position of the STA.

$N_s \cdot M$. As the smoothing length increases, the noise level gets lower, which helps to narrow the peak and improve the accuracy. However, Fig. 6(a) shows that this approach increases the risk of eliminating the beam of the real path. In fact, the strongest path does not correspond to the direct path anymore, thereby leading to an AOA error of around 10 degrees. These examples show that the right smoothing parameter setting is essential and that the optimal selection depends on the noise level. The latter varies according to several factors, such as hardware, bandwidth, number of antennas and number of subcarriers. As previously mentioned, SpotFi uses fixed smoothing length, thus not optimizing the noise level which depends on the physical parameters of the experiments.

In order to address the problem presented above we notice that prior work neglected the fact that each cluster of the 2D MUSIC should represent a real path, rather than a sub-set of the real paths, or more paths than the real ones. It follows that our objective is as follows:

- tune the factors K and J such that the *total number of clusters in the 2D MUSIC spectrum is equal to the total number of real paths presented in one packet*.

For this purpose, we use a well-known estimator of the number of paths based on CSI measurements called Matrix Pencil Method (MPM) algorithm [67], that takes as input the CSI matrix per packet and it gives as output the estimate of the number of paths. Using MPM, we model the multipath profile in the time domain as a weighted sum of delayed impulse functions, for each antenna $m \in [1, M]$, let L_m be the number of delayed paths, and $\tau_{l,m}$ and $h_{l,m}$ the propagation delay and the complex gain of the l-th path, respectively. MPM uses the CSI values as input, it operates on the frequency response of the channel and it calculates the estimate $\hat{L}_m, \hat{\tau}_{l,m}$ and $\hat{h}_{l,m}$. We then calculate the number of paths as the most frequent value Mo (the mode) of $\hat{L}$ as:

$$\hat{L} = Mo(\hat{L}_1, \hat{L}_2, .., \hat{L}_M).$$

MUSIC and MPM have been seen as antagonists in prior work, as both methods could be used to estimate channel parameters. However, in our experience, the CSI matrix used by MPM cannot be easily generalized to accommodate values from sub-carriers as it can be done with MUSIC. On the other hand, MPM shows robust performance in terms of number of paths estimation, without needing any configuration setting. Therefore, we propose to leverage both MUSIC and MPM in this work.

Using the proposed approach, in both Figs. 5(b) and 6(b), we estimate $\hat{L} = 2$ paths, so K and J are tuned in order to appreciate only two beam amplitudes above 80% of the maximum peak (the beam pattern is shown on the right of each figure), with the other peaks having a lower amplitude with respect to the peak of the direct path. Moreover, in both cases, the AOA error of the proposed multipath-based Smoothing Length is marginal. Finally, these two figures highlight that our method is the best compromise for estimating the real angle and removing the noise.

As shown in the previous examples, we propose an algorithm for tuning the smoothing factors K and J in order to obtain a total number of paths equal to the one estimated by MPM. The proposed algorithm first applies the 2D MUSIC to the first packet with the maximum smoothing factors: $K = M$ and $J = N_s$. If the estimated number of paths is less than the number of paths estimated by MPM, the algorithm iteratively decreases the smoothing resolution, by decreasing the factors K and J until the MPM number of paths is obtained. In order to avoid a very high number of possible smoothing lengths (N_{SL}), we divide the smoothing lengths into equal intervals from a maximum of $K = M$ and $J = N_s$ to a minimum of $K = (\frac{M}{2} + 1)$ and $J = \frac{N_s}{2}$. N_{SL} is selected in order to have a trade-off between

Algorithm 1: K and J Tuning Procedure.

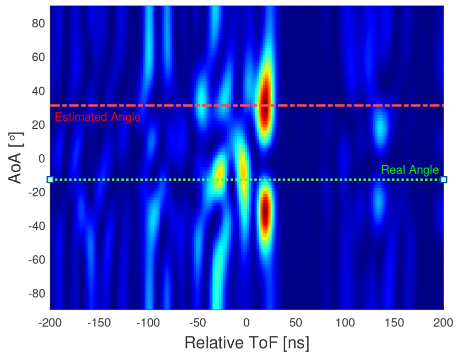
input: $\overrightarrow{CSI}$, N_{SL}, Num_packet, $Previus\_r$;
output: K, J, r;
$\overrightarrow{K}_{TMP}$ = Equidistant descending integer vector with N_{SL} values in $[M, \ldots, (\frac{M}{2}+1)]$;
$\overrightarrow{J}_{TMP}$ = Equidistant descending integer vector with N_{SL} values in $[N_s, \ldots, \frac{N_s}{2}]$;
$\hat{L}$ = number of path estimated by MPM($\overrightarrow{CSI}$);
if $Num\_packet == 1$ **then**
  $Index\_SF\_search = (1, \ldots, N_{SL})$
else
  $Index\_SF\_search$ = the one estimated by the previous packet ($Previus\_r$) and its adjacent ones
end if
for $r = Index\_SF\_search$ **do**
  $[\overrightarrow{AoA}, \overrightarrow{ToF}]$ = 2D-MUSIC($\overrightarrow{CSI}$) with resolution $\overrightarrow{K}_{TMP}(r)$ and $\overrightarrow{J}_{TMP}(r)$;
  $\hat{N}_P$ = size($\overrightarrow{AoA}$);
  if $\hat{N}_P \geq \hat{L}$ || $(r == numel(Index\_SF\_search)$ **then**
    $K = \overrightarrow{K}_{TMP}(r)$;
    $J = \overrightarrow{J}_{TMP}(r)$;
    exit;
  end if
end for

accuracy and computational cost. Furthermore, taking into consideration that the channel over time does not change dramatically for consecutive packets, for each new packet we limit the smoothing factor search to the one estimated by the previous packet and its adjacent ones. This leads to an application of 2D MUSIC at a maximum of 3 times and makes the algorithm much more computationally efficient compared to applying a complete smoothing factor search. Algorithm 1 outlines how the parameters K and J are tuned. More details about the complexity of the algorithm will be given in the Section VII-E.

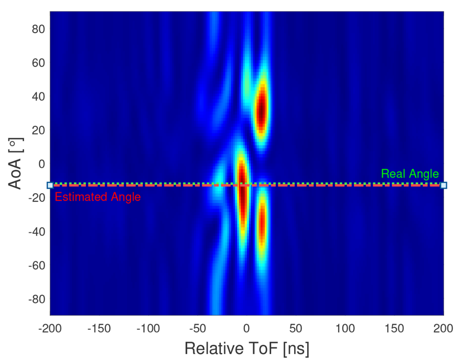
The smoothing matrix optimization could be affected by the accuracy of the MPM algorithm. In order to circumvent this problem, potential outliers are removed by the clustering algorithm and the final choice of the appropriate cluster through a likelihood function. Both operations will be described in Section IV-D.

C. Understanding Multipath Estimator in the Smoothing Matrix

MPM takes into consideration the propagation delay $\tau_{l,M}$ in order to estimate the number of paths, by estimating one path per time window. However, MPM does not have any knowledge of the AOA of each path. Therefore, it could happen that more than one path is received in the same time window from a different AOA. However, they cannot be distinguished by MPM. We show an example of this problem in Fig. 7. We have that MPM estimates $\hat{L} = 2$ paths and the real angle is equal to -13 degrees. As we can see in Fig. 7(a), optimizing the smoothing length to have two clusters with a high normalized amplitude value, we make the wrong decision in the AOA estimation. This

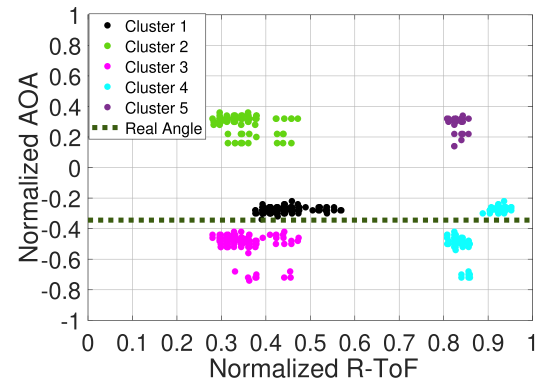


(a) 2D MUSIC spectrum with a smoothing length that has not been optimized by the MPM adjustment.

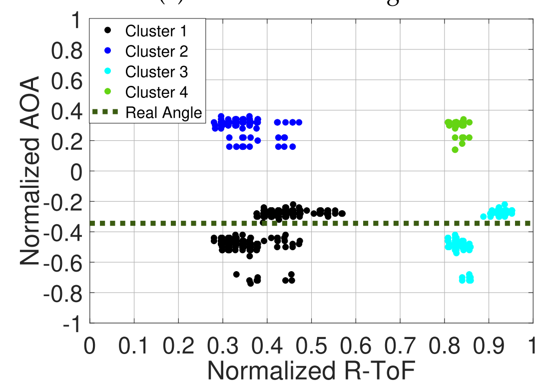


(b) 2D MUSIC spectrum with a smoothing length that has been optimized by the MPM adjustment.

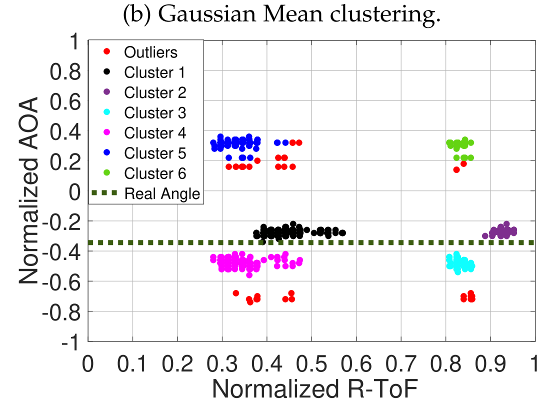
Fig. 7. 2D MUSIC spectrum and the importance of the correct interpretation of MPM estimation.



(a) K-Means clustering.



(b) Gaussian Mean clustering.



(c) Density-based spatial clustering of applications with noise.

Fig. 8. Example of application for three different clustering algorithms.

means that in this case, we have erroneously tuned the smoothing length. We then modify the methodology in order to obtain a smoothing length that corresponds to the number of paths $\hat{L}$, taking into consideration only clusters in the 2D MUSIC with different R-ToF (Fig. 7(b)). Applying this modification, as we can see in this example, we obtain a strong peak that corresponds to the real angle, thereby enhancing the procedure of tuning the smoothing length.

D. *All Together*

Referring to Fig. 4, we present in this section the remaining steps of the AOA algorithm. After smoothing the CSI matrix with the optimization presented in Sections IV-B and IV-C, we sanitize the R-ToF to eliminate the effects of Sampling Time Offset (STO). In fact, STO adds an additional phase offset that is the same across antennas for a particular subcarrier and varies over time [68]. As a result of the sanitization algorithm, the modified CSI phase response does not change even if the STO changes, making it free from the variance of changing STO.

After the R-ToF sanitization, we apply MUSIC in two dimensions AOA and R-ToF. On the left of Fig. 6, we can see two 2D MUSIC spectrums and how their resolutions change with different smoothing lengths. For each packet, we then estimate a pair of AOA and R-ToF per path, as the peak of the 2D MUSIC spectrum. Doing so, over all of the N total packets and estimating $\hat{L}$ paths per packet, we finally collect $\hat{L} \cdot N$ pairs of AOA and R-TOF estimates in the "path detection step", that are clustered together in a two-dimensional space, as shown in Fig. 8.

We then perform the clustering using all N packets. We show a representative example for clustering all the points with K-Means in Fig. 8(a), Gaussian Mean (GM) in Fig. 8(b), the method suggested in [12], and Density-Based Spatial Clustering of Application with Noise (DBSCAN) algorithm in Fig. 8(c). We choose DBSCAN for several reasons. First, DBSCAN does not need the number of clusters as input, making it a dynamic algorithm. Moreover, DBSCAN is less sensitive to the shape of the clusters, thereby managing to identify clusters that have a different shape than a circle or a sphere. Finally, as shown in the example, it is able to detect outliers (red points in Fig. 8(c)), estimating the best correct number of clusters. Finally, we assign a likelihood estimate for each cluster similarly to [12]. We declare the path with the highest likelihood metric as the direct path and store its estimated AOA ($\widehat{AOA}$). Fig. 8 shows all the estimated pairs AOA and R-ToF over 300 packets of a real case. We observe that the highest likelihood among all the clusters estimated by DBSCAN, is obtained for the cluster 1 (black cluster in Fig. 8(c)). The direct path has an angle of -35 degrees and the mean of the cluster 1 is -30 degrees. Therefore, in this example, we estimate the direction of the direct path with an error of 5 degrees.

Following the aforementioned procedure, our AoA First Path estimator gives one AoA estimate over the total N number of packets. Based on the fact that this limits the efficiency of the estimator in a practical scenario, the next section introduces a heuristic that deals with this problem.

E. Moving Windows Implementation

It is important for every estimator to give one estimate (in our case AoA estimate) per packet. As for the AoA First Path estimator, this depends on the number of packets it needs to give consistent AoA results. We observe that our estimator gives consistent AoA estimates after clustering 16 packets. Moreover, in order to obtain one AoA estimate per packet our estimator applies moving windows of $w = 16$ packets. Thus, for every new packet whose estimates are clustered (after the sixteenth packet), the estimates of the oldest one are removed. This method of moving windows helps us both to obtain one AoA estimate per packet (after the first sixteen packets) and also to understand better the impact of the channel (by extracting old measurements). An analysis of the convergence of the AoA First Path estimator, justifying the choice of 16 packets, is given in Section VII-E.

To sum up, our AoA First Path estimator initially leverages the number of propagation paths, estimated by MPM, to perform an optimized and dynamic CSI smoothing (Sections IV-B and IV-C). Then, it applies the R-ToF sanitization algorithm to eliminate the effects of STO. After that, the application of 2D MUSIC estimates the AoA and R-ToF pairs and a clustering of the estimated AoA and R-ToF follows, based on DBSCAN. The cluster that has the highest likelihood is the one whose AoA is chosen. Finally, the above mentioned procedure is applied using moving windows of w packets.

We stress that the R-ToF studied in this section is only used for estimating the AOA through the likelihood function. After the application of the sanitization algorithm, the effects of varying STOs are removed, but still this R-ToF is not the real (absolute) one and cannot be used for ranging purposes [12]. For this reason, the ranging purposes FTM protocol is used as studied in the next section.

V. Fine Time Measurements

This section introduces the FTM protocol of the IEEE 802.11 standard and the proposed model for the FTM noise detection.

A. IEEE 802.11mc Background

IEEE 802.11 standardized the FTM protocol to estimate the distance between a pair of WiFi chipsets. An FTM initiator (FTMI) is a STA that initiates the FTM process by sending an FTM Request to the AP. An AP that supports the FTM procedure as a responding device is called a responder (FTMR). If the FTMR agrees to start the measurements, it sends an FTM message to the FTMI and waits for its acknowledgement (ACK). The Round Trip Time (RTT) is calculated taking into consideration both the transmission timestamp of the FTM message and also the reception timestamp of its ACK. In the computation, the protocol subtracts the time that the STA needs to send back the ACK from the total RTT.

B. FTM Sources of Noise

Signal propagation in rich indoor environments is subject to multipath effects, where multiple coherent copies of the transmitted signal arrive at the receiver over different reflected paths. It is even possible that the direct component is severely attenuated and the signal is received mostly over reflected paths. Since signals that travel over reflected paths will take longer time to arrive at the receiver, they introduce an error in the distance estimation when considering the A-ToF. We define the following function y for l-th path:

$$y = \log_{10}(d_l) + \mathcal{N}(0, \sigma_{\mathcal{N}}) \quad d_l \geq d_0. \tag{4}$$

$\mathcal{N}(0, \sigma_{\mathcal{N}})$ represents an additive Gaussian noise $\mathcal{N}$, with a standard deviation $\sigma_{\mathcal{N}}$ and d_0 is equal to 1 m. This expression is inspired by the path loss model with the log-normal distribution that represents the shadowing effect.

Now, let $\mathcal{S}$ be the set of samples. We express the generic FTM sample as:

$$d_{l,s} \quad l \in L, s \in \mathcal{S}. \tag{5}$$

Based on the model in (4), in Section V-C we introduce the first path estimator f to mitigate the effects of the main sources of noise in the channel. The estimator will operate in the log domain, i.e., $log_{10}(d_{l,s})$. The ranging system may give $d_{l,s}$ smaller than d_0. As such, we add a constant factor for the purpose of operating in the log domain, such that it is larger than d_0 for a sequence of samples.

C. First Path Estimator for FTM

This subsection describes in detail the proposed first path estimator for FTM.

In (5), each FTM sample $d_{l,s}$ is affected by a distance bias caused by the absence or presence of multipath. Grouping together the samples with the same bias results in a finite Gaussian Mixture Model (GMM) in the log domain with a small number of modes. One of these modes corresponds to the samples received through the direct path, while the others correspond to the samples received through any of the reflected paths. Then, knowing the number of estimated paths, we can separate all the Gaussian components and the median or the mean of the first Gaussian would be a reliable estimator of the direct path's distance. For this purpose, we exploit the CSI measurements, and we rely on the MPM algorithm, introduced in Section IV-B.

Fig. 9 shows an example of a real case with a travelled distance of the direct path equal to 10.73 m. Supposing the MPM output is unknown, the estimator f estimates the parameters of the components as follows: the means of the first Gaussian are

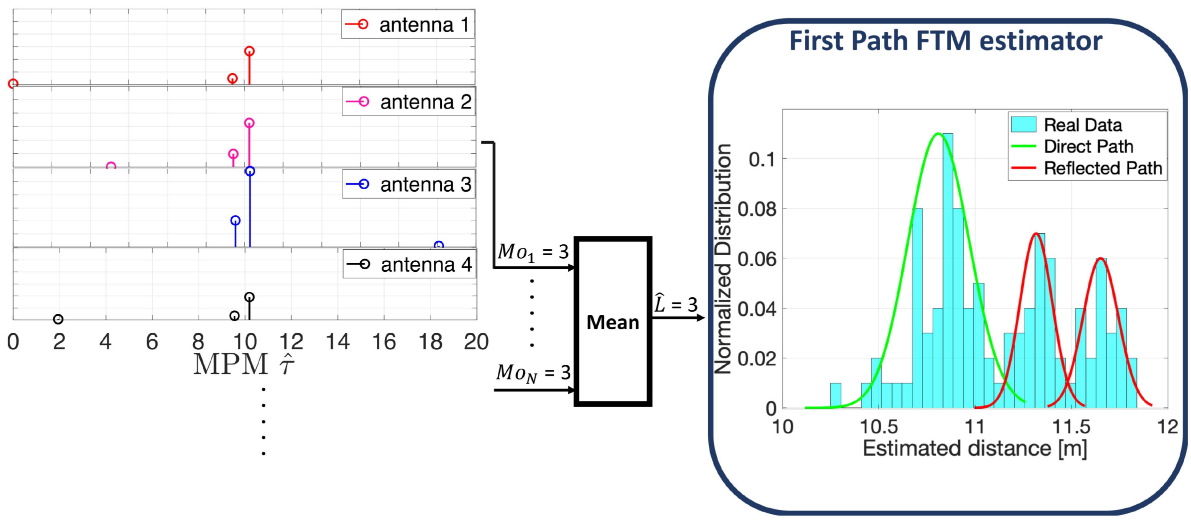


Fig. 9. Example of the first path estimator f for a real case where the distance between AP and target is 10.73 m. The number of paths, Mo_1, is the first most frequent value of the estimations provided by the MPM for each antenna and the mean ($\hat{L}$) over all N packets is given as input to the first path estimator f, which after that computes only the mean of the first log-Gaussian.

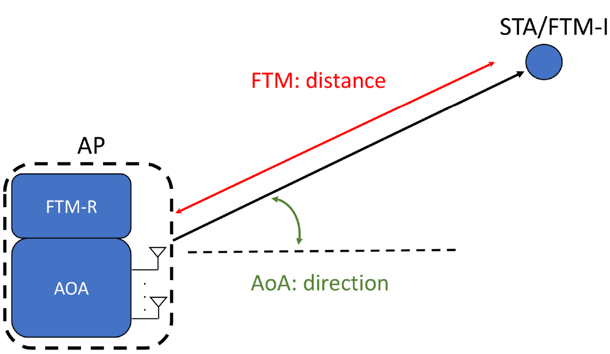


Fig. 10. FTM/AOA approach for smartphone positioning.

equal to 11.23, 11.01, 10.82 and 10.6 meters for an estimated number of paths from 1 to 4, respectively. As MPM provides an estimated number of paths equal to 3, it follows that we estimate the distance of the direct path with an over-estimation error of only 0.09 m. In Section VII we show the performance of the proposed filter in four different testbeds.

VI. System Deployment

We first present the experimental platform used in this work and then the testbeds used for the evaluation.

A. Experimental Platform

Our approach for smartphone positioning is shown in Fig. 10, and the commodity hardware we use as the AP is shown in Fig. 3. It is composed of a multi-antennas AP which enables the collection of Multiple Input Multiple Output (MIMO) CSI, and operation as FTM-R.

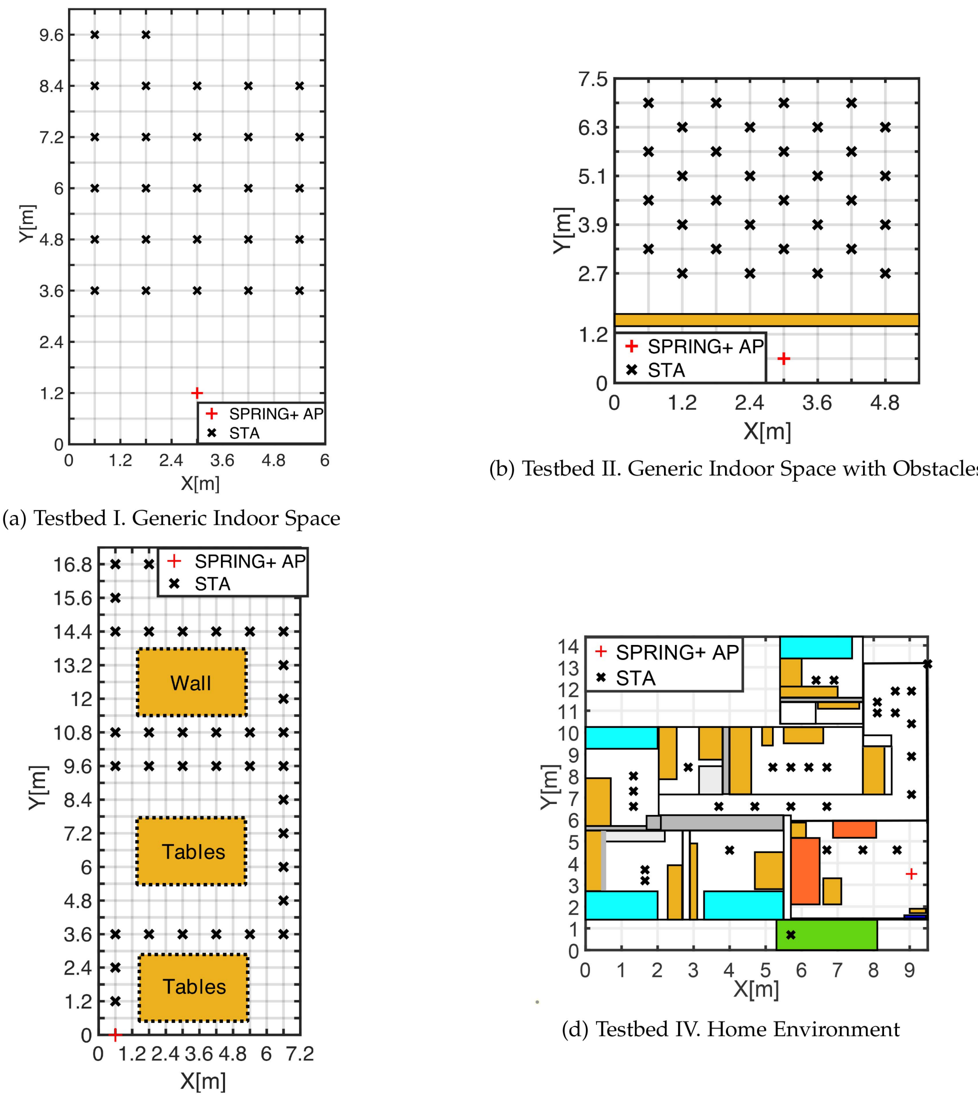
In order to collect CSI, we use a QHS8405S4-RDK device, the Quantenna (QTNA) 4×1 Uniform Linear Array (ULA). QTNA supports PCIe, RGMII and 802.11a/n/ac protocol. The frequency range is from 5.15 GHz to 5.85 GHz and it supports 20/40/80 MHz bandwidth. QTNA enables rapid collection of precise high-order MIMO CSI. The spatial diagnostics interface is supported on QTNA's BBIC4 based platform and it supports extracting up to 4×1 channels with bandwidth up to 80 MHz, with CSI data from the driver accessed over a Transmission Control Protocol (TCP) socket. Any WiFi device can be used as the STA.

Regarding the FTM protocol, we use the fitlet2-CJ3455 platform as responder FTM-R since it is an integrated solution in compact form, which includes the WiFi Intel 8265 chipset. We use the WiFi Indoor Location Device (WILD) tool for configuring the FTM-R [69].

As a STA, we use the Google Pixel 3 phone with Android Pie (API Level 28) that supports the FTM protocol. The phone operates as FTM-I for time measurements. The device must have location-based services enabled at the system level to access the FTM protocol. We use the android-WifiRttScan application to initiate the measurements. We modify its code to facilitate the data collection, and we configure it to receive a distance measurement per packet. Its main activity lists all of the APs using the WifiManager. By selecting an AP that supports FTMR, another activity is launched and a RangingRequest is initiated via the WifiRttManager. The activity displays and stores many of the details returned from the FTMR including the distance reported between the AP and the smartphone.

B. Deployment Scenarios

We perform experiments in four indoor testbeds, namely Testbed I, II, III and IV. The first one represents a *generic indoor space*, the second one a *generic indoor space with obstacles*, the third one an *office environment* and the fourth, referred as a *home environment*, is a real home in Madrid city. The maps are shown in Fig. 11. Testbed I, in Fig. 11(a), covers a surface of almost 65 m^2. We use 27 selected locations (marked as crosses) to test our system, and the propagation is mainly over a Line-Of-Sight (LOS) path. Deploying a single AP, the number of links is equal to the number of target STA locations. Furthermore, Testbed II



(a) Testbed I. Generic Indoor Space

(b) Testbed II. Generic Indoor Space with Obstacles

(c) Testbed III. Office Environment

(d) Testbed IV. Home Environment

Fig. 11. Testbeds to assess the direction, ranging and positioning capabilities of SPRING+.

is depicted in Fig. 11(b), it covers a space of around 40 m^2 and the target device is placed in 32 different locations. In Testbed II, the propagation is over Non Line-Of-Sight (NLOS), since it contains a concrete wall (yellow rectangle in Fig. 11(b)) between the AP and the STA locations. Moreover, Testbed III can be seen in Fig. 11(c), it covers a space of around 125 m^2 and the target device is placed in 35 different positions. In Testbed III, the propagation takes place through a mixture of LOS and NLOS paths. Testbed III contains several obstacles, such as concrete walls and tables (yellow boxes in Fig. 11(c)) and it is surrounded by glasses. Our proposed fourth testbed is a real house shown in Fig. 11(d), that covers an area of around 140 m^2 and includes 30 target devices. As shown with different colors in Fig. 11(d), Testbed IV includes two long corridors, obstacles (such as walls and doors) and a wide range of furniture (e.g. tables, desks, beds), which act as reflectors. All experiments are conducted with other active WiFi networks in the neighborhood. Both CSI and FTM measurements are obtained on a fixed frequency channel in the 5 GHz band. For the evaluation we use a single AP (“SPRING+ AP”, red marker in Fig. 11). Both the Access Point (AP) and the STA are in the same height of 1 m. We deploy “SPRING+ AP” as an AP and the Google Pixel 3 smartphone as target STA in all marked positions, shown in Fig. 11. For each testing location we gather 300 data samples.

VII. Evaluation

In this section, we first analyze the deployed testbeds and then the performance of the proposed methods for computing direction, range and finally positioning of the STA in Testbed I, II, III and IV. We deploy the Google Pixel 3 smartphone as target STA in all marked positions, shown in Fig. 11.

A. Estimated Paths in Each Testbed

In this subsection, we first motivate the choice of the MPM algorithm for the estimation of the number of paths, and we then use the MPM for the evaluation of the four deployed testbeds complexity.

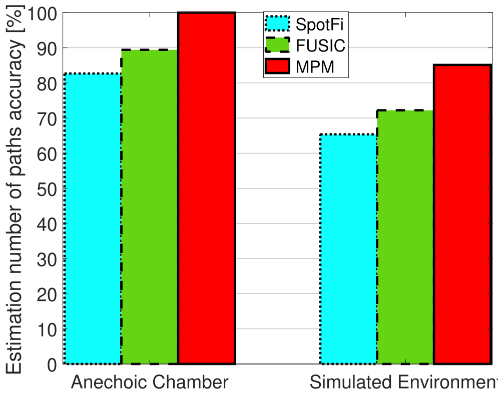


Fig. 12. Estimation paths accuracy in two different controlled scenarios.

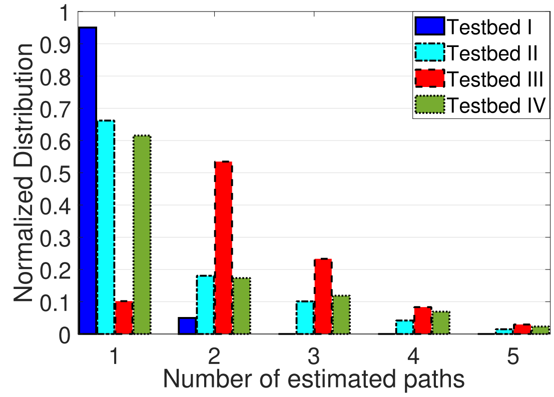


Fig. 13. Normalized histograms of the number of estimated paths for all four Testbeds.

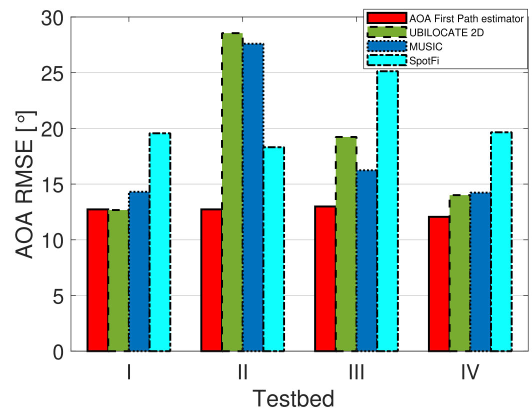


Fig. 14. AOA RMSE in degrees for AOA First Path estimator, MUSIC, UBILOCATE 2D and SpotFi.

For this purpose, we start investigating the performance of the MPM algorithm in two different controlled environments. As seen in Section IV-B, MPM uses the CSI values as input, it operates on the frequency response channel, and it calculates the estimated number of paths per antenna. In order to evaluate the accuracy of the MPM algorithm, we perform a study in two scenarios for which the number of paths is known: an experimental anechoic chamber and a simulated environment. In the first scenario, it is reasonably safe to assume a number of paths equal to one, while the simulated environment allows us to control the reflections at the receiver, thus fixing the total number of received paths.

Anechoic Chamber: We fix the STA in the middle of the room and we rotate the AP in order to span an angle of π, from $-\pi/2$ to $+\pi/2$, where 0° corresponds to the normal direction of the antenna array elements, which are placed as Uniform Linear Array (ULA) with a distance of $\lambda/2$ between antennas. We conduct 36 experiments, every 5°, and for each of them we collect hundreds of CSI samples.

Simulated Environment: We use MATLAB, setting all the network parameters, such as the main frequency, the bandwidth, the number of subcarriers and the modulation, compatible with the real experiments described in Section VI-A. For each simulated packet, the number of paths is randomly selected between one and five [9]. For each path, we randomly set the Signal to Noise Ratio (SNR) and the attenuation parameters.

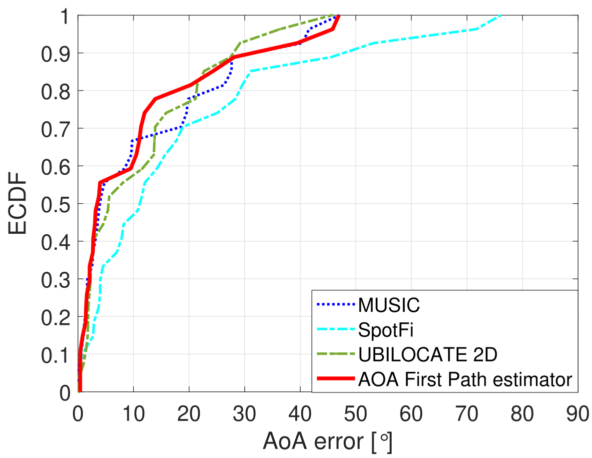
We summarize the results of the accuracy in percentile on the estimation of the number of paths in Fig. 12. MPM achieves an accuracy of 100% and 85.11% in the anechoic chamber and in the simulated environment, respectively. We compare the obtained results with both SpotFi [12] and FUSIC [61]. As discussed in Section IV-B, SpotFi is able to estimate the number of paths and their relative pairs AOA and R-ToF, fixing the smoothing length and the threshold for the detection of the peaks in the 2D MUSIC spectrum. FUSIC first calculates the number of peaks in the MUSIC spectrum and then removes the peaks with relative strength, compared to the main peak, below a certain threshold. The number of paths is then estimated as the number of the filtered peaks. We observe in the figure that the maximum accuracy is obtained by the MPM algorithm. In fact, SpotFi and FUSIC approaches achieve an accuracy of 89.41% and 82.67% in the anechoic chamber and 72.21% and 65.36% in the simulated environment.

We finally highlight the difference in the deployed scenarios showing in Fig. 13 the number of estimated paths, using the MPM algorithm only. In Testbed I, MPM estimates a single path almost 100% of the time, while in Testbed II, III and IV it estimates a variable number of paths (from 1 to 5), due to the mixture of LOS and NLOS wireless links. We also observe that Testbed III has the highest number of reflections due to glasses and walls in this office environment.

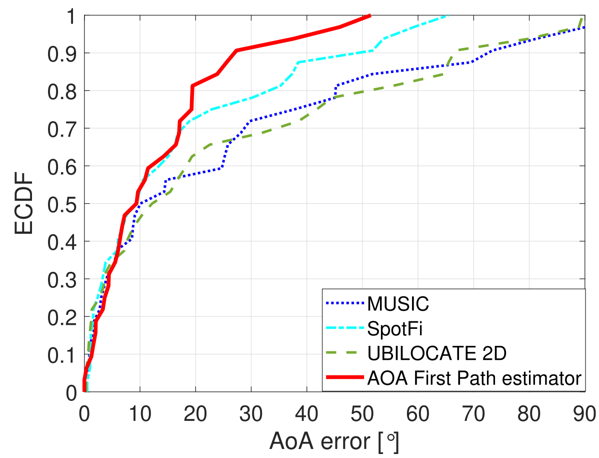
B. AOA

We collect CSI measurements from the QTNA device that communicates with a Google Pixel 3 smartphone. We estimate the AOA according to the methodology presented in Section IV, and we then evaluate the AOA estimation error in all testbeds. The map of the testbeds is used to compute the ground truth AOA for the evaluation.

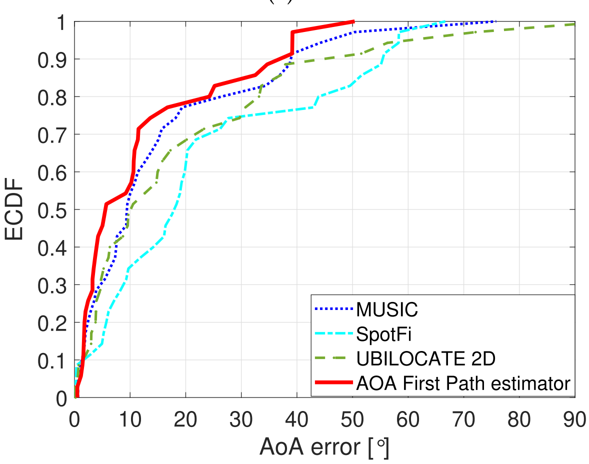
We summarize the results of the AOA Root Mean Square Error (RMSE) in degrees obtained for each of the 4 algorithms in all testbeds in Fig. 14, using four algorithms: MUSIC (used also in our preliminary work SPRING [25]), SpotFi [12], Ubilocate [51] and our proposed First Path AOA estimator. [51] is a recent



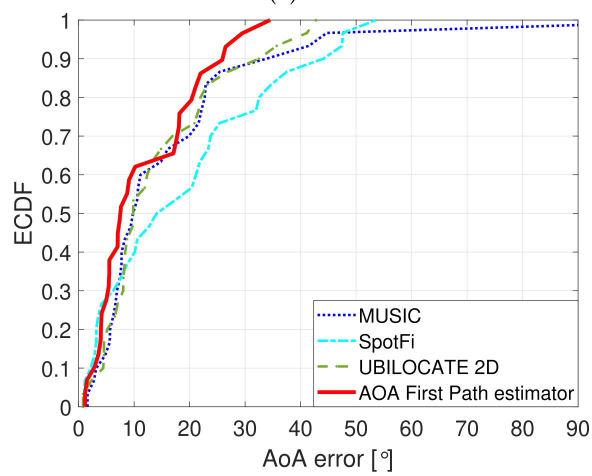
(a) Testbed I.



(b) Testbed II.



(c) Testbed III.



(d) Testbed IV.

Fig. 15. ECDF of AOA estimation error in degrees for all four Testbeds.

attempt that estimates AOA with a 2-step procedure. As a first step, it estimates the path parameters (AOA, Angle of Departure, A-ToF) of all paths. After that, it applies the Nelder-Mead Search algorithm to refine these parameters, thereby obtaining more accurate estimations. Furthermore, it introduces 2 models: one that takes into account only the AOA and ToF and another one that considers also the angle of departure. We compare our model with the first one (or Ubilocate 2D as hereafter denoted), since only this model is applicable to our data (we do not have measurements for Angle of Departure). As we can see, our proposed *AOA First Path estimator* has consistently the lowest AOA RMSE, while other estimators may perform well in some testbeds, but then fail in other deployments.

In Fig. 15, we then study the Empirical Cumulative Distribution Functions (ECDFs) of the AOA error in degrees. We observe that, in the LOS Testbed I (Fig. 15(a)), we have a median error of the proposed SPRING+ AOA estimator of around 3.5°, while MUSIC, Ubilocate 2D and SpotFi achieve a median error of 4°, 5.5° and 11° respectively. In a completely NLOS testbed, namely Testbed II, we see from Fig. 15(b) that our First Path AOA estimator presents a median (80-percentile) error of 9.3 (19)°, while SpotFi, MUSIC and Ubilocate 2D achieve errors of 9.3 (35)°, 10 (45)° and 12.1 (50)°respectively. Moving to Testbed III, Fig. 15(c) shows that our estimator clearly outperforms again the other 3 models both with regards to the median error (5.5°) and the 80-th percentile (24°). More specifically, MUSIC and Ubilocate 2D share similar behavior with a median error of 9.5° and 80-th percentile of around 33°, whereas SpotFi is weaker with corresponding errors equal to 18° and 43°. In this testbed, both our proposed solution and MUSIC are able to detect efficiently the first path even if this testbed is the one with the highest number of paths (cf. Fig. 13). Finally, as shown in Fig. 15(d), in the real house Testbed IV, our estimator outperforms the other three algorithms, achieving a median error of 8°, compared to the median error of MUSIC, Ubilocate 2D and SpotFi, which is around 10° for the first two models and 14° for the last one.

Furthermore, we observe that SpotFi obtains worse results than MUSIC on three out of four available Testbeds. As explained in Section IV-B, our intuition is that SpotFi, fixing the smoothing length, does not optimize the noise level with the physical parameters of our experiments. For this reason, SpotFi

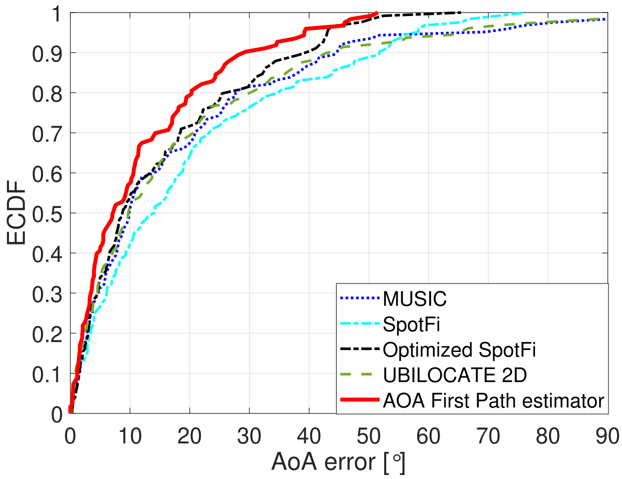


Fig. 16. Aggregated ECDF of AOA error in degrees among all Testbeds.

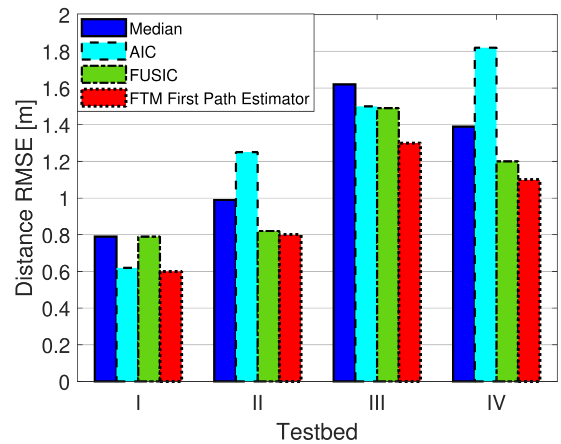


Fig. 17. Distance RMSE in meters for our estimator (red bars) compared to the median (blue bars), AIC (cyan bars) and FUSIC (green bars).

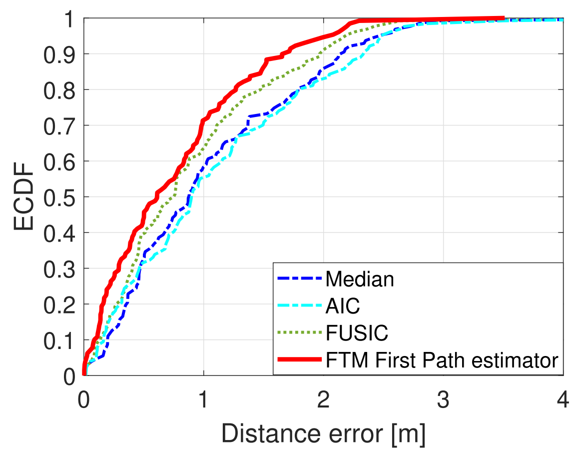


Fig. 18. Aggregated ECDF of distance error in meters among all Testbeds.

increases the risk of estimating false paths, most of the time resulting in ECDFs outperformed by MUSIC. In order to verify our intuition, we present an aggregated AoA error ECDF among all of our four testbeds, including the models described above and also an optimized version of SpotFi. In fact, we apply the best smoothing parameters $K = 4$ and $J = 200$, chosen a-posteriori among possible N_{SL} Smoothing Factors. This model is called "Optimized SpotFi" and its results can be seen in Fig. 16. In other terms, this shows that an initial calibration of smoothing parameters is necessary since as seen in the figure, this optimized version of SpotFi performs better than MUSIC. Besides, the Optimized SpotFi version is still worse than our estimator, which does not fix the smoothing parameters for all the experiments, but dynamically varies these for each CSI received based on MPM estimation. This result confirms the efficiency of the proposed dynamic smoothing algorithm (Algorithm 1), which is an important competitive advantage of our AoA First Path estimator.

Concluding, our estimator is robust across different environments, in comparison with the other prior algorithms.

C. Distance

Following the study on AOA, in this subsection we investigate the ranging performance. Fig. 17 shows the ranging Root Mean Square Error (RMSE) in meters for Testbed I, II, III and IV. We compare the obtained results with three different solutions: the median, the Akaike Information Criterion (AIC) and FUSIC [61]. The latter estimates the distance between the AP and the STA correcting the raw FTM estimate with an excess delay provided by the CSI values. More specifically, it first calculates the number of peaks in the MUSIC spectrum, and then their relative strengths compared to the main peak. If the relative strength is higher than a certain threshold, FUSIC keeps the initial FTM measurement, otherwise the distance estimation is calculated by subtracting the mean excess delay from the raw FTM measurement. The AIC is commonly applied to identify the optimal number of clusters in GMM. We use the lowest AIC to infer the optimal number of paths [70], and then we use the path with the least positive mean as ranging estimate. We observe in the figure that the minimum ranging error is obtained by the proposed FTM First Path estimator, achieving a ranging gain of 18%, 26% and 12% with respect to the median, AIC and FUSIC, respectively. Furthermore, in Fig. 18 we show the aggregated ECDF of the ranging errors among all of our 4 testbeds including the aforementioned models. As illustrated in Fig. 18, our proposed FTM First estimator outperforms the alternatives consistently. We highlight that in terms of distance estimation, we do not make a comparison with Ubilocate [51] or SpotFi [12], since such a comparison is not applicable to our hardware or data. As for Ubilocate, they introduce a distance estimation protocol which is implemented in their firmware and it is not standard compliant. As for SpotFi, they do not estimate A-ToF.

D. Positioning

In this subsection, we study the localization error. We define a coordinate system on a two-dimensional map. Considering a single AP system, let (x_{AP}, y_{AP}) be the position of the AP, $\hat{d}$ the estimate of the distance from the AP to the target and $\hat{\theta}$ the estimated direction between the AP and the target. We find the

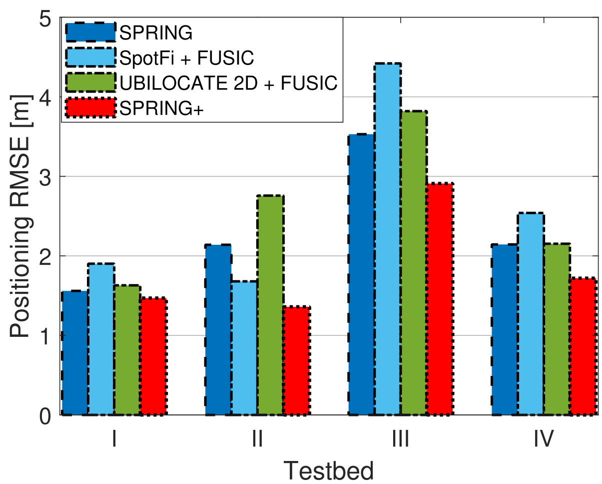


Fig. 19. Positioning RMSE in meters for all four Testbeds.

estimated coordinates of the STA as follows:

$$\hat{p} = (\hat{x}, \hat{y}) = (x_{AP} + \hat{d} \cdot \cos\hat{\theta}, y_{AP} + \hat{d} \cdot \sin\hat{\theta}). \quad (6)$$

We study the position accuracy of our proposed system SPRING+. SPRING+ is composed of the AoA First Path estimator for the direction (AoA) estimate and the FTM First Path estimator for the distance estimate. We compare the obtained results with three different systems. The first one is our preliminary work SPRING [25], that uses MUSIC to identify the strongest AOA, and the FTM First Path estimator to estimate the distance. Furthermore, we construct two other models consisted either of SpotFi [12] or Ubilocate 2D [51] for the AoA estimate and FUSIC for the distance estimate. The last two models are, hereafter, denoted as "SpotFi + FUSIC" and "Ubilocate 2D + FUSIC" respectively.

We summarize the results in Fig. 19, where we show the positioning RMSE in meters obtained for each algorithm in all testbeds. We observe that the RMSE of SPRING+ is lower than the RMSE of the other algorithms in all four testbeds. More in detail, we highlight an average gain in positioning error of 21 %, 28 % and 29 % with respect to SPRING, "Ubilocate 2D + FUSIC" and "SpotFi + FUSIC" respectively among the four different testbeds.

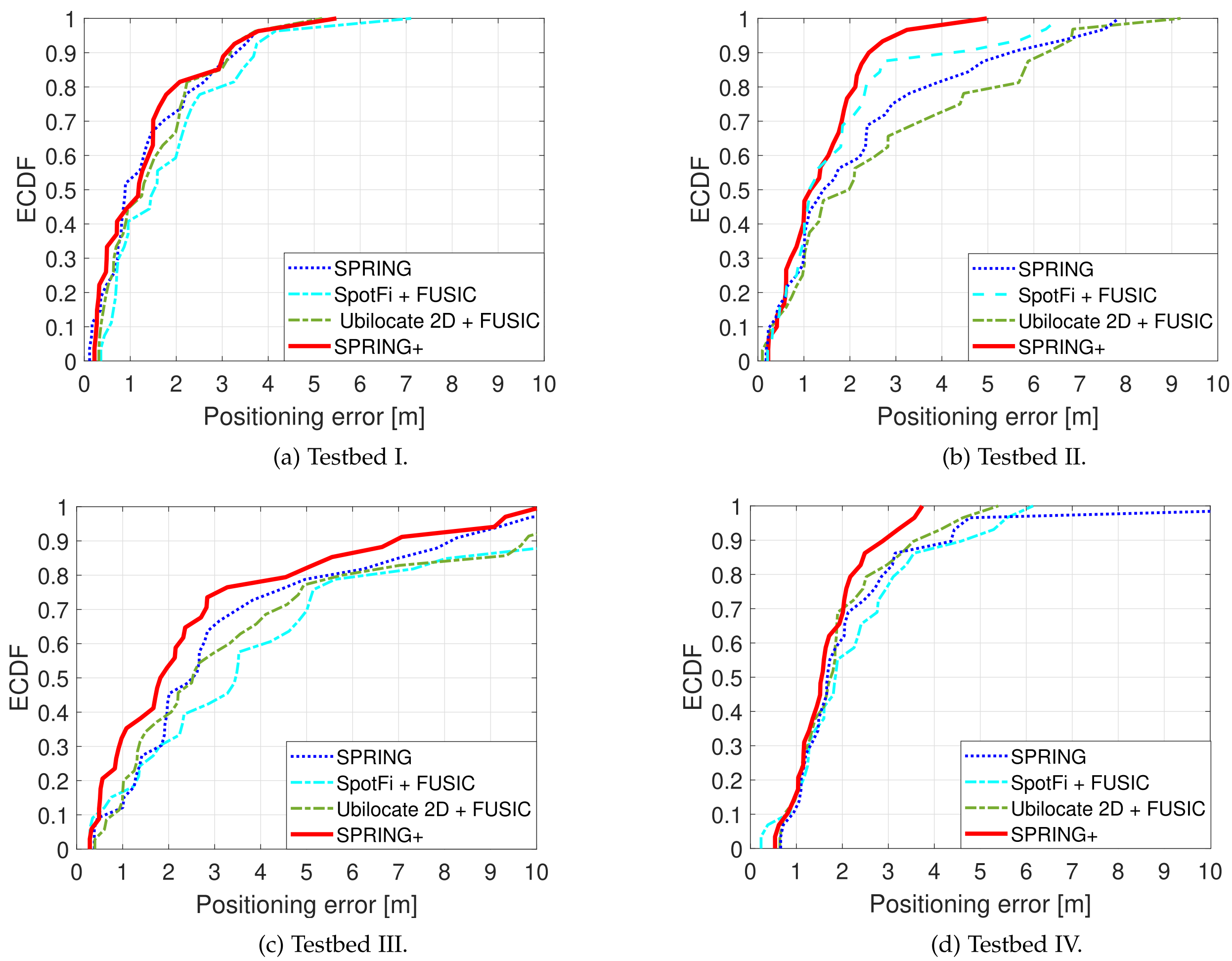
Finally, Fig. 20 shows the ECDFs of the positioning error obtained by our proposed system SPRING+ and the above-mentioned systems in the evaluated four testbeds. The figure shows that SPRING+ achieves a median error between 1 and 1.8 meters and the 80-percentile positioning error in the range of 1.9–4.6 meters, thereby providing superior performance to state-of-the-art approaches.

E. Stability and Time Complexity

Time complexity and stability play an important role for every algorithm. As for the stability, we plot the AoA RMSE with a varying length of moving windows between 10 and 50 packets. As we see in the Fig. 21, the AoA RMSE of our proposed estimator converges after using a maximum moving window of 16 CSI packets in all of our four Testbeds. This justifies the choice of 16 packets moving windows (Section IV-E). This optimal value was calculated using testbeds that cover a wide range of indoor environments with different complexities (LOS, NLOS, mixed) and can be utilized for secure reproduction of our proposed estimator.

As for the time complexity, we highlight that the most computationally heavy part of the AoA First Path estimator is the application of 2D MUSIC (for a maximum of three times, see Section IV-B) for obtaining the dynamic Smoothing Factor. More in detail, the main source of time complexity of 2D MUSIC algorithm comes from the eigenvalue matrix decomposition and is equal to $O((M * N_s)^3)$, where M is the number of antennas and N_s is the number of subcarriers [71]. The same is also mentioned as the time complexity of SpotFi model [72], which is logical since SpotFi applies the 2D MUSIC algorithm too. As for MUSIC algorithm, the time complexity for the matrix eigendecomposition is $O(M^3)$, where M is again the number of antennas [71]. Furthermore, as for Ubilocate [51], the time complexity comes mainly from the application of Nelder- Mead search algorithm and is equal to $O(M * N_s * P)$, where M, N_s represent the number of antennas and subcarriers respectively and P the number of iterations required for convergence. Finally, as shown in Section V-B, the distance (ranging) estimate is provided in one step using (4). and its complexity in the overall system is negligible, so the FTM First Path estimator does not add any computational overhead to our system. Based on this outlook, the total computational complexity of SPRING+ corresponds to the computational complexity of the AoA estimator which is equal to $O((M * N_s)^3)$.

As for the execution time, it is also dependent on the programming language that we use. For this reason, we have written the 2D MUSIC and dynamic Smoothing Length estimation into a C language code. The C code shows a more clear picture of the execution time of our estimator in a real implementation. The median time complexity of our AoA First Path Estimator for one Smoothing Factor is 0.22 seconds, using a PC with a processor Intel(R) Core(TM) 15 i7-8700 K CPU with 3.70 GHz and a RAM of 16 GB and a single core. As discussed in the Section IV-B, the Smoothing Factor choice needs an application of 2-D MUSIC up to three times, thus proportionally increasing the aforementioned execution time. Based on this outlook, an implementation of our estimator in a PC with 3 cores is recommended that can execute the 2-D MUSIC algorithm for the Smoothing Factor choice in parallel. Furthermore, as already mentioned in Section VII-E, our proposed distance estimator (FTM First Path estimator) has negligible impact on the computational overhead of our positioning system, so no other core is needed for it. Therefore, after completing the training phase of 16 packets, SPRING+ can estimate the target STA location after 0.22 seconds using a PC with similar characteristics as the one described above but with just 3 cores. We note that these results are performed with non-optimized code. Therefore, any code optimization will further improve the results. As for the other algorithms, an exact execution time comparison is not possible. SpotFi does not estimate one AoA per packet, but per total number of clustered packets. Based on this, they are faster than our estimator for one



(a) Testbed I. (b) Testbed II.

(c) Testbed III. (d) Testbed IV.

Fig. 20. ECDF of Positioning error in meters for all four Testbeds.

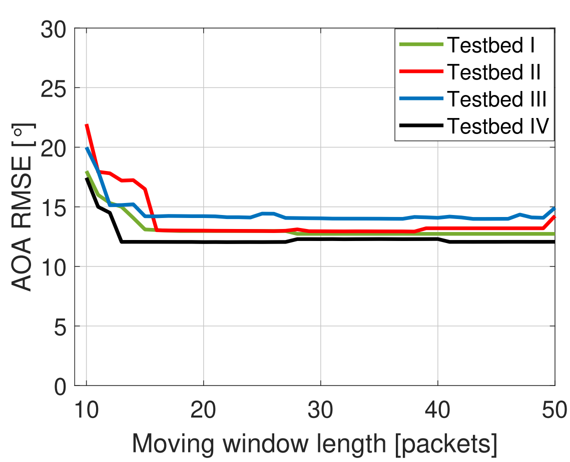


Fig. 21. AoA RMSE per moving window length for all Testbeds.

packet, since they apply 2D MUSIC only once with the minimum smoothing factors ($117x3$), whereas we apply 2D MUSIC up to a maximum of 3 times with smoothing factors usually larger than the minimum one. However, in the end SpotFi needs more time for the final estimation than our AoA First Path estimator, since it has to cluster all the available packets to obtain one AoA estimate, whereas we obtain an estimate using moving windows of 16 packets. As for Ubilocate and MUSIC, considering the time complexity analysis that was made before, we can verify that they are less computationally expensive than our estimator in compensation for their weakest accuracy.

Concluding, our AoA First Path estimator does not have a prohibitive execution time and can be useful for near real-time applications.

VIII. Conclusion

In this paper we presented SPRING+, an indoor positioning system that requires a single access point to localize commercial off-the-shelf smartphones with high accuracy. We experimentally demonstrated the feasibility to position a smartphone through WiFi measurements performed by a single AP using commodity hardware. The solution leverages on measurements collected from an 802.11ac AP with 4 linear antennas, that operates at 80 MHz and has access to CSI per sub-carrier and FTM data per packet. We used this information to design a method able to estimate the first path for angle and distance measurements. We highlighted how critical the multipath estimator is for the construction of the dynamic CSI Smoothing Length and how our system is able to deal with noisy measurements in the 2-D MUSIC Spectrum, thereby improving the AoA accuracy. In terms of ranging, the proposed First Path estimator again used the information provided by the multipath estimator, thus managing to obtain reliable results. Using both direction and distance estimates, SPRING+ demonstrated its indoor localization effectiveness in an extensive experimental campaign comprising four different testbeds including generic, office and home environments. Our results show that SPRING+ is able to achieve a median 2D positioning error of a commodity smartphone between 1 and 1.8 meters with a single WiFi AP.

Finally, our system can be applicable to other well-known problems. The techniques presented in this work can also be applied to the traditional case of positioning using multiple APs. Furthermore, Multipath profile analysis, using 2D MUSIC enhanced by the Smoothing Length optimization for both LOS and NLOS scenarios, can be useful for tasks such as passive localization, human tracking or contact tracing. Exploring these directions of research is a part of our future work. SPRING+ code will be released as open source.

References

[1] "TS 38.215, User Equipment (UE) performance requirements for Radio Access Technology (RAT) Independent Positioning Enhancements," in 3rd Generation Partnership Project, Release 17, 2020. Accessed: Nov., 2023. [Online]. Available: https://www.3gpp.org/

[2] "TS 37.171, user equipment (UE) performance requirements for rat-independent positioning enhancements." in 3rd Generation Partnership Project, Release 17, 2022. Accessed: Nov., 2023. [Online]. Available: https://www.3gpp.org/

[3] P. Bahl and V. Padmanabhan, "RADAR: An in-building RF-based user location and tracking system," in *Proc. Conf. Comput. Commun., 19th Annu. Joint Conf. IEEE Comput. Commun. Societies*, 2000, pp. 775–784.

[4] M. Youssef and A. Agrawala, "The horus WLAN location determination system," in *Proc. 3rd Int. Conf. Mobile Syst. Appl. Serv.*, 2005, pp. 205–218, doi: 10.1145/1067170.1067193.

[5] H. Lim, L.-C. Kung, J. Hou, and H. Luo, "Zero-configuration, robust indoor localization: Theory and experimentation," in *Proc. 25th IEEE Int. Conf. Comput. Commun.*, 2006, pp. 1–12.

[6] A. Goswami, L. E. Ortiz, and S. R. Das, "WiGEM: A learning-based approach for indoor localization," in *Proc. Seventh Conf. Emerg. Netw. Experiments Technol.*, 2011, pp. 3:1–3:12, doi: 10.1145/2079296.2079299.

[7] K. Chintalapudi, A. Padmanabha Iyer, and V. N. Padmanabhan, "Indoor localization without the pain," in *Proc. 16th Annu. Int. Conf. Mobile Comput. Netw.*, 2010, pp. 173–184, doi: 10.1145/1859995.1860016.

[8] A. Rai, K. K. Chintalapudi, V. N. Padmanabhan, and R. Sen, "Zee: Zero-effort crowdsourcing for indoor localization," in *Proc. 18th Annu. Int. Conf. Mobile Comput. Netw.*, 2012, pp. 293–304, doi: 10.1145/2348543.2348580.

[9] J. Xiong and K. Jamieson, "ArrayTrack: A fine-grained indoor location system," in *Proc. 10th USENIX Symp. Networked Syst. Des. Implementation*, 2013, pp. 71–84. [Online]. Available: https://www.usenix.org/conference/nsdi13/technical-sessions/presentation/xiong

[10] S. Sen, J. Lee, K.-H. Kim, and P. Congdon, "Avoiding multipath to revive inbuilding Wi-Fi localization," in *Proc. 11th Annu. Int. Conf. Mobile Syst. Appl. Serv.*, New York, NY, USA: ACM, 2013, pp. 249–262, doi: 10.1145/2462456.2464463.

[11] J. Gjengset, J. Xiong, G. McPhillips, and K. Jamieson, "Phaser: Enabling phased array signal processing on commodity WiFi access points," in *Proc. 20th Annu. Int. Conf. Mobile Comput. Netw.*, 2014, pp. 153–164.

[12] M. Kotaru, K. Joshi, D. Bharadia, and S. Katti, "SpotFi: Decimeter level localization using WiFi," in *Proc. ACM Conf. Special Int. Group Data Commun.*, New York, NY, USA: ACM, 2015, pp. 269–282, doi: 10.1145/2785956.2787487.

[13] X. Li, K. Pahlavan, M. Latva-aho, and M. Ylianttila, "Comparison of indoor geolocation methods in DSSS and OFDM wireless LAN systems," in *Proc. Veh. Technol. Conf.*, 2000, pp. 3015–3020.

[14] D. D. McCrady, L. Doyle, H. Forstrom, T. Dempsey, and M. Martorana, "Mobile ranging using low-accuracy clocks," *IEEE Trans. Microw. Theory Techn.*, vol. 48, no. 6, pp. 951–958, Jun. 2000.

[15] A. Marcaletti, M. Rea, D. Giustiniano, V. Lenders, and A. Fakhreddine, "Filtering noisy 802.11 time-of-flight ranging measurements," in *Proc. 10th ACM Int. Conf. Emerg. Netw. Experiments Technol.*, New York, NY, USA: ACM, 2014, pp. 13–20, doi: 10.1145/2674005.2674998.

[16] D. Giustiniano and S. Mangold, "CAESAR: Carrier sense-based ranging in off-the-shelf 802.11 wireless LAN," in *Proc. 7th Conf. Emerg. Netw. Experiments Technol.*, New York, NY, USA: ACM, 2011, pp. 10:1–10:12, doi: 10.1145/2079296.2079306.

[17] D. Vasisht, S. Kumar, and D. Katabi, "Decimeter-level localization with a single WiFi access point," in *Proc. 13th USENIX Symp. Networked Syst. Des. Implementation*, Santa Clara, CA: USENIX Association, 2016, pp. 165–178. [Online]. Available: https://www.usenix.org/conference/nsdi16/technical-sessions/presentation/vasisht

[18] Z. Chen et al., "AWL: Turning spatial aliasing from foe to friend for accurate WiFi localization," in *Proc. 13th Int. Conf. Emerg. Netw. EXperiments Technol.*, New York, NY, USA: ACM, 2017, pp. 238–250, doi: 10.1145/3143361.3143377.

[19] A. T. Mariakakis, S. Sen, J. Lee, and K.-H. Kim, "SAIL: Single access point-based indoor localization," in *Proc. 12th Annu. Int. Conf. Mobile Syst. Appl. Serv.*, New York, NY, USA: ACM, 2014, pp. 315–328, doi: 10.1145/2594368.2594393.

[20] K. Qian, C. Wu, Y. Zhang, G. Zhang, Z. Yang, and Y. Liu, "Widar2. 0: Passive human tracking with a single Wi-Fi link," in *Proc. 16th Annu. Int. Conf. Mobile Syst. Appl. Serv.*, 2018, pp. 350–361.

[21] K. Liu, Z. Tian, Z. Li, J. Wang, and M. Zhou, "HiLoc: Sub-meter level indoor localization using a single access point with distributed antennas in wireless sensor networks," *IEEE Sensors J.*, vol. 22, no. 6, pp. 4869–4881, Mar. 2022.

[22] E. Gönültaş, E. Lei, J. Langerman, H. Huang, and C. Studer, "CSI-based multi-antenna and multi-point indoor positioning using probability fusion," *IEEE Trans. Wireless Commun.*, vol. 21, no. 4, pp. 2162–2176, Apr. 2022.

[23] D. Sánchez-Rodríguez, M. A. Quintana-Suárez, I. Alonso-González, C. Ley-Bosch, and J. J. Sánchez-Medina, "Fusion of channel state information and received signal strength for indoor localization using a single access point," *Remote Sens.*, vol. 12, no. 12, 2020, Art. no. 1995.

[24] IEEE, 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 802.11–2020 (Revision of IEEE Standard 802.11-2016), 2021.

[25] M. Rea, T. E. Abrudan, D. Giustiniano, H. Claussen, and V.-M. Kolmonen, "Smartphone positioning with radio measurements from a single Wi-Fi access point," in *Proc. 15th Int. Conf. Emerg. Netw. Experiments Technol.*, New York, NY, USA: Association for Computing Machinery, 2019, pp. 200–206, doi: 10.1145/3359989.3365427.

[26] K. Wu, J. Xiao, Y. Yi, M. Gao, and L. M. Ni, "FILA: Fine-grained indoor localization," in *Proc. IEEE INFOCOM*, 2012, pp. 2210–2218.

[27] H. Zhang et al., "MetaRadar: Indoor localization by reconfigurable metamaterials," *IEEE Trans. Mobile Comput.*, vol. 21, no. 8, pp. 2895–2908, Aug. 2022.

[28] B. Yang, L. Guo, R. Guo, M. Zhao, and T. Zhao, "A novel trilateration algorithm for RSSI-based indoor localization," *IEEE Sensors J.*, vol. 20, no. 14, pp. 8164–8172, Jul. 2020.

[29] Y. Li et al., "Cost-effective localization using RSS from single wireless access point," *IEEE Trans. Instrum. Meas.*, vol. 69, no. 5, pp. 1860–1870, May 2020.

[30] H. Zhang, H. Zhang, B. Di, K. Bian, Z. Han, and L. Song, "Metalocalization: Reconfigurable intelligent surface aided multi-user wireless indoor localization," *IEEE Trans. Wirel. Commun.*, vol. 20, no. 12, pp. 7743–7757, Dec. 021.

[31] M. Azizyan, I. Constandache, and R. Roy Choudhury, "Surroundsense: Mobile phone localization via ambience fingerprinting," in *Proc. 15th Annu. Int. Conf. Mobile Comput. Netw.*, 2009, pp. 261–272.

[32] Z. Yang, C. Wu, and Y. Liu, "Locating in fingerprint space: Wireless indoor localization with little human intervention," in *Proc. 18th Annu. Int. Conf. Mobile Comput. Netw.*, 2012, pp. 269–280.

[33] H. Liu et al., “Push the limit of WiFi based localization for smartphones,” in *Proc. 18th Annu. Int. Conf. Mobile Comput. Netw.*, 2012, pp. 305–316.
[34] H. Wang, S. Sen, A. Elgohary, M. Farid, M. Youssef, and R. R. Choudhury, “No need to war-drive: Unsupervised indoor localization,” in *Proc. 10th Int. Conf. Mobile Syst. Appl. Serv.*, 2012, pp. 197–210.
[35] R. Nandakumar, K. K. Chintalapudi, and V. N. Padmanabhan, “Centaur: Locating devices in an office environment,” in *Proc. 18th Annu. Int. Conf. Mobile Comput. Netw.*, 2012, pp. 281–292.
[36] Z. Gao, Y. Gao, S. Wang, D. Li, and Y. Xu, “CRISLoc: Reconstructable CSI fingerprinting for indoor smartphone localization,” *IEEE Internet Things J.*, vol. 8, no. 5, pp. 3422–3437, Mar. 2021.
[37] X. Tong, Y. Wan, Q. Li, X. Tian, and X. Wang, “CSI fingerprinting localization with low human efforts,” *IEEE/ACM Trans. Netw.*, vol. 29, no. 1, pp. 372–385, Feb. 2021.
[38] A. Foliadis and M. H. C. Garcia, R. A. Stirling-Gallacher, and R. S. Thomä, “CSI-based localization with CNNs exploiting phase information,” in *Proc. IEEE Wirel. Commun. Netw. Conf.*, 2021, pp. 1–6.
[39] X. Wang, X. Wang, and S. Mao, “Deep convolutional neural networks for indoor localization with CSI images,” *IEEE Trans. Netw. Sci. Eng.*, vol. 7, no. 1, pp. 316–327, First Quarter 2020.
[40] T. Koike-Akino, P. Wang, M. Pajovic, H. Sun, and P. V. Orlik, “Fingerprinting-based indoor localization with commercial MMWave WiFi: A deep learning approach,” *IEEE Access*, vol. 8, pp. 84 879–84 892, 2020.
[41] M. Abbas, M. Elhamshary, H. Rizk, M. Torki, and M. Youssef, “WiDeep: WiFi-based accurate and robust indoor localization system using deep learning,” in *Proc. IEEE Int. Conf. Pervasive Comput. Commun.*, 2019, pp. 1–10.
[42] H. Rizk, A. Elmogy, and H. Yamaguchi, “A robust and accurate indoor localization using learning-based fusion of Wi-Fi RTT and RSSI,” *Sensors*, vol. 22, no. 7, 2022, Art. no. 2700.
[43] A. Kokkinis, L. Kanaris, A. Liotta, and S. Stavrou, “RSS indoor localization based on a single access point,” *Sensors*, vol. 19, no. 17, 2019, Art. no. 3711.
[44] R. Ayyalasomayajula et al., “Deep learning based wireless localization for indoor navigation,” in *Proc. 26th Annu. Int. Conf. Mobile Comput. Netw.*, 2020, pp. 1–14.
[45] Y. Zheng, J. Liu, M. Sheng, and C. Zhou, “Exploiting fingerprint correlation for fingerprint-based indoor localization: A deep learning-based approach,” in *Machine Learning for Indoor Localization and Navigation*. Berlin, Germany: Springer, 2023, pp. 201–237.
[46] C. Peng, G. Shen, Z. Han, Y. Zhang, Y. Li, and K. Tan, “A beepbeep ranging system on mobile phones,” in *Proc. 5th Int. Conf. Embedded Networked Sensor Syst.*, 2007, pp. 397–398.
[47] S. Kumar, S. Gil, D. Katabi, and D. Rus, “Accurate indoor localization with zero start-up cost,” in *Proc. 20th Annu. Int. Conf. Mobile Comput. Netw.*, 2014, pp. 483–494.
[48] S. Sen, J. Lee, K.-H. Kim, and P. Congdon, “Avoiding multipath to revive inbuilding WiFi localization,” in *Proc. 11th Annu. Int. Conf. Mobile Syst. Appl. Serv.*, 2013, pp. 249–262.
[49] K. Joshi, S. Hong, and S. Katti, “Pinpoint: Localizing interfering radios,” in *Proc. 10th USENIX Symp. Networked Syst. Des. Implementation*, 2013, pp. 241–253.
[50] D. Niculescu and B. Nath, “VOR base stations for indoor 802.11 positioning,” in *Proc. 10th Annu. Int. Conf. Mobile Comput. Netw.*, 2004, pp. 58–69.
[51] A. Blanco et al., “Accurate ubiquitous localization with off-the-shelf IEEE 802.11 AC devices,” in *Proc. 19th Annu. Int. Conf. Mobile Syst. Appl. Serv.*, 2021, pp. 241–254.
[52] Y. Xie, J. Xiong, M. Li, and K. Jamieson, “mD-Track: Leveraging multidimensionality for passive indoor Wi-Fi tracking,” in *Proc. 25th Annu. Int. Conf. Mobile Comput. Netw.*, 2019, pp. 1–16.
[53] M. Heydariaan, H. Dabirian, and O. Gnawali, “Anguloc: Concurrent angle of arrival estimation for indoor localization with UWB radios,” in *Proc. 16th Int. Conf. Distrib. Comput. Sensor Syst.*, 2020, pp. 112–119.
[54] Y. Hou, X. Yang, and Q. H. Abbasi, “Efficient AoA-based wireless indoor localization for hospital outpatients using mobile devices,” *Sensors*, vol. 18, no. 11, 2018, Art. no. 3698.
[55] A. Blanco, P. J. Mateo, F. Gringoli, and J. Widmer, “Augmenting mmWave localization accuracy through sub-6 GHz on off-the-shelf devices,” in *Proc. 20th Annu. Int. Conf. Mobile Syst. Appl. Serv.*, 2022, pp. 477–490.
[56] Z. Chen et al., “M 3: Multipath assisted Wi-Fi localization with a single access point,” *IEEE Trans. Mobile Comput.*, vol. 20, no. 2, pp. 588–602, Feb. 2021.
[57] X. Tong, H. Li, X. Tian, and X. Wang, “Wi-Fi localization enabling self-calibration,” *IEEE/ACM Trans. Netw.*, vol. 29, no. 2, pp. 904–917, 2021.
[58] W. Gong and J. Liu, “SiFi: Pushing the limit of time-based WiFi localization using a single commodity access point,” *Proc. ACM InterAct. Mobile Wearable Ubiquitous Technol.*, vol. 2, no. 1, pp. 1–21, 2018.
[59] D. Vasisht, S. Kumar, and D. Katabi, “Decimeter-level localization with a single Wi-Fi access point,” in *Proc. 13th USENIX Symp. Netw.ed Syst. Des. Implementation*, 2016, pp. 165–178.
[60] A. Poulose and D. S. Han, “UWB indoor localization using deep learning LSTM networks,” *Appl. Sci.*, vol. 10, no. 18, 2020, Art. no. 6290.
[61] K. Jiokeng, G. Jakllari, A. Tchana, and A.-L. Beylot, “When FTM discovered MUSIC: Accurate WiFi-based ranging in the presence of multipath,” in *Proc. IEEE Conf. Comput. Commun.*, 2020, pp. 1857–1866.
[62] T. Otim, A. Bahillo, L. E. Díez, P. Lopez-Iturri, and F. Falcone, “Towards sub-meter level UWB indoor localization using body wearable sensors,” *IEEE Access*, vol. 8, pp. 178 886–178 899, 2020.
[63] T. Otim, L. E. Díez, A. Bahillo, P. López-Iturri, and F. Falcone, “Effects of the body wearable sensor position on the UWB localization accuracy,” *Electronics*, vol. 8, no. 11, p. 1351, 2019.
[64] X. Liu et al., “Kalman filter-based data fusion of Wi-Fi RTT and PDR for indoor localization,” *IEEE Sensors J.*, vol. 21, no. 6, pp. 8479–8490, Mar. 2021.
[65] H. Jin and P. Papadimitratos, “Off-the-shelf Wi-Fi indoor smartphone localization,” in *Proc. 17th Wirel. On-Demand Netw. Syst. Serv. Conf.*, 2022, pp. 1–4.
[66] R. Schmidt, “Multiple emitter location and signal parameter estimation,” *IEEE Trans. Antennas Propag.*, vol. 34, no. 3, pp. 276–280, Mar. 1986.
[67] Y. Hua and T. K. Sarkar, “Matrix pencil method for estimating parameters of exponentially damped/undamped sinusoids in noise,” *IEEE Trans. Acoust. Speech Signal Process.*, vol. 38, no. 5, pp. 814–824, May 1990.
[68] S. Sen, B. Radunovic, R. R. Choudhury, and T. Minka, “You are facing the Mona Lisa: Spot localization using PHY layer information,” in *Proc. 10th Int. Conf. Mobile Syst. Appl. Serv.*, 2012, pp. 183–196.
[69] “WiFi indoor location device (WILD) software,” Accessed: Aug. 2022. [Online]. Available: http://fit-pc.com/wiki/index.php/WiFi_Indoor_Location_Device_(WILD)_software
[70] S. Konishi and G. Kitagawa, *Information Criteria and Statistical Modeling*. Berlin, Germany: Springer, 2008.
[71] Y.-Y. Wang, J.-T. Chen, and W.-H. Fang, “TST-MUSIC for joint DOA-delay estimation,” *IEEE Trans. Signal Process.*, vol. 49, no. 4, pp. 721–729, Apr. 2001.
[72] W. Gong and J. Liu, “RoArray: Towards more robust indoor localization using sparse recovery with commodity WiFi,” *IEEE Trans. Mobile Comput.*, vol. 18, no. 6, pp. 1380–1392, Jun. 2019.

Stavros Eleftherakis is currently working toward the PhD degree with the Pervasive Wireless Systems Group, Imdea Networks Research Institute (Madrid, Spain) and University Carlos III of Madrid (UC3M). His main research interests include positioning systems, network security, and privacy and artificial intelligence.

Giuseppe Santaromita (Member, IEEE) is postdoc researcher with IMDEA Networks Institute, within the Pervasive Wireless Systems Group. His research activities are focused on wireless networks. In particular on the programmable PHY layer for optimizing wireless networks performance and on the low-latency and high-accuracy localization methods, mainly on 5 G New Radio networks.

Maurizio Rea received the PhD degree in telematics engineering from the University Carlos III of Madrid. He is senior researcher with i2CAT Foundation, Barcelona, Spain. His interests include data analysis, wireless communication, mmWave networks, beamforming algorithms, channel state information, angle of arrival algorithms, and context-aware mechanisms.

Xavier Costa-Pérez (Senior Member, IEEE) received the MSc and PhD degrees in telecommunications from the Polytechnic University of Catalonia (UPC). He is currently a research professor with ICREA, scientific director with i2Cat and the head of 6G R&D with NEC Laboratories Europe. His team generates research results which are regularly published at top scientific venues, produces innovations which are the recipient of several awards for successful technology transfers, participates in major European Commission R&D collaborative projects and contributes to standardization bodies.

Domenico Giustiniano (Senior Member, IEEE) is a research associate professor with IMDEA Networks Institute and leader of the Pervasive Wireless System Group. He also worked for a total of four years in industrial research labs (Disney Research and Telefonica Research). He is co-founder of LiFi4Food, a startup for precision agriculture, and leader of the OpenVLC project. His current research interests include cover battery-free IoT systems, large-scale spectrum-based analytics, and 5G+ localization.