Wi-Fi-Based Real-Time UAV Localization: A Comparative Analysis Between RSSI-Based and FTM-Based Approaches

EMANUELE PAGLIARI
Luleå University of Technology, Luleå, Sweden

LUCA DAVOLI, Member, IEEE
GIANLUIGI FERRARI, Senior Member, IEEE
University of Parma, Parma, Italy
National Inter-University Consortium for Telecommunications (CNIT), Research Unit of Parma, Parma, Italy

Wi-Fi connectivity for localization purposes has been used for several years in the Internet of Things (IoT) context, where the (general) static nature of IoT devices allows to approximately localize them in known environments with low effort and implementation costs. While the accuracy of Wi-Fi localization for IoT applications can be considered as acceptable, the adoption of Wi-Fi-based localization for (a highly mobile) unmanned aerial vehicle (UAV) has received limited attention. In this article, a low-cost and low-complexity system architecture is proposed and exploited to perform a comparative analysis between two Wi-Fi-based localization approaches: the traditional received signal strength indicator (RSSI) ranging and the more recent fine time measurement (FTM), based on the IEEE 802.11mc amendment. Our goal is to estimate and compare the efficacy of the proposed system for real-time positioning of a static or moving UAV, evaluating the impact of different filtering solutions on the localization accuracy. The obtained results show that FTM-based localization is more accurate, reducing the positioning error by 37% with respect to the RSSI-based positioning approach. Our results also confirm the better overall performance of the FTM-based solution for low-cost localization applications, discussing its limitations, scalability, and advantages as a viable backup positioning solution in (weak or denied) Global Navigation Satellite System-based environments and scenarios.

Manuscript received 8 February 2024; revised 11 July 2024; accepted 21 July 2024. Date of publication 25 July 2024; date of current version 6 December 2024.

DOI. No. 10.1109/TAES.2024.3433829

Refereeing of this contribution was handled by J. Seo.

This work was supported by the European Union's Horizon 2020 research and innovation program ECSEL Joint Undertaking (JU) through ADACORSA project—"Airborne Data Collection on Resilient System Architectures" under Grant 876019 and in part by the European Union's Horizon Europe research and innovation program Key Digital Technology (KDT) JU through OPEVA project—"OPtimization of Electric Vehicle Autonomy" under Grant 101097267.

Authors' addresses: Emanuele Pagliari was with the IoT Lab of the Department of Engineering and Architecture, University of Parma, Parma, 43124 Parma, Italy. He is now with the Robotics and AI Group, Department of Computer, Electrical and Space Engineering, Luleå University of Technology, 97187 Luleå, Sweden, E-mail: (emanuele.pagliari@ltu.se); Luca Davoli and Gianluigi Ferrari are with the Internet of Things (IoT) Lab, Department of Engineering and Architecture, University of Parma, 43124 Parma, Italy, and also with the National Inter-University Consortium for Telecommunications (CNIT), Research Unit of Parma, 43124 Parma, Italy, E-mail: (luca.davoli@unipr.it; gianluigi.ferrari@unipr.it). *(Corresponding author: Luca Davoli.)*

NOMENCLATURE

List of the Acronyms Adopted in this Article

ACK	Acknowledgment.
AGL	Above ground level.
AoA	Angle of arrival.
AP	Access point.
BVLOS	Behind visual line-of-sight.
COTS	Commercial-off-the-shelf.
CSI	Channel state information.
ECEF	Earth-centered Earth-fixed.
ECDF	Empirical cumulative distribution function.
EKF	Extended Kalman filter.
EMA	Exponential moving average.
FC	Flight controller.
FTM	Fine time measurement.
GNSS	Global navigation satellite system.
IMU	Inertial measurement unit.
IoT	Internet of Things.
JSON	JavaScript object notation.
KF	Kalman filter.
LOS	Line-of-sight.
LS	Least squares.
LSMR	Least squares with multiple right.
MAC	Media access control.
MIMO	Multiple-input–multiple-output.
ML	Machine learning.
NLOS	Non-line-of-sight.
PDR	Pedestrian dead reckoning.
RADAR	Radio detection and ranging.
RF	Radio frequency.
RPi4	Raspberry Pi 4.
RSSI	Received signal strength indicator.
RTT	Round-trip time.
SBC	Single board computer.
SSID	Service Set IDentifier.
ToA	Time of arrival.
ToD	Time of departure.
ToF	Time of flight.
TSME	Two-step M-estimator.
TWR	Two-way ranging.
UAV	Unmanned aerial vehicle.
UWB	Ultra wideband.
VSLAM	Visual simultaneous localization and mapping.

WGS84 World geodetic system 1984.

I. INTRODUCTION

Radio localization has been investigated from various perspectives. In fact, given the widespread adoption of wireless connectivity, exploiting already available infrastructures for low-cost asset tracking purposes has always been very attractive [1]. To this end, efforts have been spent in the development of low-complexity and affordable localization solutions, especially relying on existing Wi-Fi infrastructures (typically available in industrial and public environments). However, while in the literature (as will be shown in Section II) several results have been proposed for Internet of Things (IoT) applications, where a position error on the order of several meters can be considered sufficient to roughly estimate the position of an IoT device (e.g., in a specific area of a warehouse), the same does not hold for mobile robotic platforms, such as unmanned aerial vehicles (UAVs). In this case, in fact, the combination of the mobile nature of the platform together with the need to achieve the best possible accuracy (ideally a position estimation error not higher than 1 m), especially in critical behind visual line-of-sight (BVLOS) applications, makes the application of existing approaches challenging and extremely expensive.

Submeter localization accuracy can already be obtained through the use of ultra wideband (UWB) technology [2], which allows to achieve a position estimation error not higher than 30 cm [3], especially in combination with a dense infrastructure composed by several UWB anchors deployed in the environment. However, the presence of this infrastructure significantly increases the initial deployment cost and the complexity needed to keep all the deployed anchors synchronized with each other for localization purposes. Moreover, such high accuracy might not be needed in all the applications involving robotic platforms: for instance, for some missions, a position estimation error of a few meters, as already provided by traditional Global Navigation Satellite System (GNSS) systems, is sufficient. Therefore, in order to keep the initial deployment cost as low as possible, as well as to allow a localization accuracy of a few meters in medium-to-large environments, the use of radio technologies different from UWB must be investigated.

In this article, we first investigate existing Wi-Fi-based localization systems, with the aim to predict the achievable performance in both static and mobile conditions. Then, targeting the development of a low-cost real-time localization system for UAVs and comparing two different Wi-Fi-based localization techniques, we propose a novel localization architecture suitable for unmanned drones, detailing its composing modules and the corresponding processing algorithms (designed to maximize the positioning accuracy). In particular, the proposed system allows to carry out a comparative analysis of two possible Wi-Fi-based localization approaches: 1) traditional received signal strength indicator (RSSI)-based approach, and 2) more recent fine time measurement (FTM)-based approach. Then, localization capabilities and positioning errors in both static and mobile conditions are provided, together with an experimental performance comparison between the chosen approaches. In addition, the use of proper filtering approaches is investigated. Our goal is to show the viability of Wi-Fi-based localization as a backup solution suitable to provide an approximate real-time position of a UAV flying in a GNSS-denied environment. This is beneficial to allow the pilot, while remotely controlling the UAV flying in BVLOS conditions and in the absence of GNSS, to understand in which part of the predefined flight path the UAV is operating. In order to reach this goal, the proposed Wi-Fi-based localization system is implemented on a real UAV to experimentally validate its performance and understand if the localization accuracy achievable with Wi-Fi FTM-based localization solutions is sufficient for real-time (approximate) localization of a mobile UAV. Our experimental implementation is based on the adoption of affordable commercial-off-the-shelf (COTS) hardware operating on the 2.4-GHz band.

For the sake of clarity and convenience, in the Nomenclature, we list all the acronyms adopted in this article.

The rest of this article is organized as follows. In Section II, we overview existing literature on Wi-Fi-based technologies for localization purposes in both IoT and UAV worlds. In Section III, we detail the considered Wi-Fi-based localization techniques, while Section IV discusses the adoption of Kalman filters (KFs) for both outlier and measurement noise removal, with the aim to improve the accuracy of the position estimation. In Section V, we present the proposed Wi-Fi-based localization architecture, discussing its real-time capabilities. Preliminary experimental results of the proposed system, integrated on a real UAV platform, are presented in Section VI. Improvements of the proposed system and future research directions are discussed in Section VII. Finally, Section VIII concludes this article.

II. STATE OF THE ART

Radio-based localization is the most investigated and adopted solution for GNSS-denied environments, where several radio anchors are available (depending on the adopted radio technology). In detail, it is exploited to allow UAVs to operate not only in such critical environments, but also to detect unwanted UAVs flying nearby critical areas and infrastructures, as detailed in [4] and [5], where radio-based approaches to identify and localize an incoming UAV are discussed. Then, with regard to *on-board* radio-based localization, several approaches have been investigated, such as traditional RSSI-based ranging and fingerprinting, as well as more advanced solutions, such as time of flight (ToF) ranging. For the sake of completeness, an deeper overview and discussion is proposed in Section III.

The most traditional approach for commercial wireless communications relies on the estimation of the distances between the target and the anchors (with known positions)—corresponding to access points (APs) in Wi-Fi networks—on the basis of the RSSI. Using the Friis [6] propagation

loss model, the RSSI allows to approximately determine the distance between the transmitter and the receiver.[1] Collecting at least three distance estimations from different APs, one can perform a multilateration geometrical bidimensional position estimation of the target to be localized.

Although in the last decades several research activities (as, for example, [7] and [8]) have adopted the RSSI ranging techniques for localization purposes, their relative accuracy still remains an open issue, being affected by radio environment noise floor, external interference, and line-of-sight (LOS) or non-LOS (NLOS) signal propagation. Therefore, the accuracy of RSSI-based localization techniques purely relies on the accuracy on the underlying distance estimation model and adopted noise filtering techniques. The majority of the proposed works, such as [9], manage to achieve errors on the order of several meters in both LOS and NLOS environments using Wi-Fi RSSI localization approaches based on different signal propagation models to properly estimate the LOS or NLOS nature of the considered target–AP links. More in detail, for static objects (where multiple RSSI measurements can be gathered and the LOS/NLOS nature of each target–AP link can be characterized), it is possible to achieve a position estimation error of a few meters, as discussed in [9]. However, adopting more complex filtering solutions, as well as a combination of additional measurements and machine learning (ML) techniques (such as in [10]), the performance can be improved. For example, the solutions presented in [11], based on a Wi-Fi RSSI fingerprinting approach, lead to a position estimation error below 1 m. Another example of advanced techniques applied to Wi-Fi RSSI fingerprinting is discussed in [12], where multiple approaches are evaluated in order to determine the best algorithm suitable to achieve the highest positioning accuracy.

While the above results hold, providing satisfying positioning capabilities, for static targets, for mobile targets (e.g., flying UAVs), when only a few (or only one) RSSI measurements can be gathered from each anchor, the position estimation error can significantly increase. In order to keep it suitable to UAV localization applications, a large number of APs are required to obtain a sufficient position estimation, as shown in [13]. A real experimental evaluation of mobile target localization through RSSI measurements is discussed by Booranawong et al. in [14], where they achieved (in very specific environmental and experimental conditions) a localization error between 1 and 5 m (with respect to the ground truth), depending on the number of gathered measurement and adopted filtering techniques. Instead, in [15], an advanced sensor fusion technique for pedestrian localization, combining Wi-Fi RSSI together with pedestrian dead reckoning, is proposed to reduce the error on moving targets position tracking. Although the result is interesting, its applicability to UAV localization might be difficult.

Besides the RSSI-based distance estimation technique, more consistent and reliable approaches have appeared, mostly relying on advanced techniques requiring specific hardware-implemented features, such as, for example, the ToF measurement of the transmitted radio signals. The ToF technique allows a more accurate distance estimation between transmitter and receiver, especially when used in combination with high-frequency and high-bandwidth protocols (i.e., UWB and 5 GHz Wi-Fi). In [16], a localization technique for smartphones has been proposed exploiting the FTM mechanism of the 2.4-GHz Wi-Fi IEEE 802.11mc protocol [17], aided by the angle of arrival (AoA) [18] measured on the 5-GHz band: the obtained results show a promising positioning error below 1 m in a 8×9 m^2 region with LOS target–AP links. However, the results obtained in [16] refer to a static target—for each tested position, a large number of target–AP distance measurements are collected. Therefore, in [16], the impact of the mobility is not investigated. A similar approach (still for smartphones) is presented in [19], where FTM is used in combination with the RSSI fingerprinting technique to estimate the position of a smartphone inside a 500 m^2 area, in the end obtaining an average position estimation error below 1 m. However, this solution requires an intensive premeasurement phase, which is not always feasible. Relevant results have been achieved by a novel approach proposed in [20], where the combination of LOS and NLOS channel estimation with the use of the 5-GHz band APs allows to achieve an average position estimation error between 1 and 2 m over a 300 m^2 area on a smartphone in static and mobile conditions, respectively.

In [21], the combined utilization of Wi-Fi FTM and GNSS data, fused together through an extended Kalman filter (EKF), is proposed to reach meter-level localization accuracy in pedestrian applications, relying also on the use of an inertial measurement unit integrated in the device developed in [21]. Despite the promising results, this solution has been validated only for pedestrian applications.

Regarding the adoption of Wi-Fi FTM on UAVs, in [22], a solution able to achieve a submeter positioning error with four 5-GHz APs within a 5×5 m^2 area, with the UAV hovering in a static position, is proposed. However, the extension of the testbed is not representative of realistic applications for UAV localization. The approach proposed in [23], exploiting a hardware similar to the one adopted in this article for both RSSI- and FTM-based approaches—in detail, the 2.4-GHz affordable ESP32S2-enabled development boards used as APs—manages to obtain a 1.5-m positioning error within a 5×10 m^2 testing area (owing also to a novel ML filtering solution) in the presence of a static target to be localized. In [24] and [25], FTM measurements of Wi-Fi signals are used, together with other sensing technologies, to

[1]The Friis formula refers to unquantized transmitter and receiver powers. In real-world scenarios, various factors, such as obstacles, reflections, and absorption, can affect the signal strength as it travels through the medium, resulting in received power levels lower than those predicted by the Friis formula. Moreover, since RSSI provides a measure of the received signal's power level in a discretized form obtained from the receiver's hardware, establishing a straightforward relationship between RSSI and the distance is challenging due to the intricate nature of wireless propagation environments.

provide distance estimation for UAV applications in several scenarios, including IoT applications, research and rescue, as well as autonomous drone landing close to an FTM-enabled target. Despite the promising results, the solutions detailed in [24] and [25] do not investigate the localization performance relying on the sole FTM-based positioning is provided, both in static and moving conditions, highlighting technology, which, instead, is used as additional to enable specific missions. In [26], a comparison between Wi-Fi RSSI-based and FTM-based smartphone positioning techniques is proposed, confirming an overall reduction of the positioning error by 16% with the FTM-based approach using a 2.4 GHz 20 MHz bandwidth Wi-Fi infrastructure with respect to the RSSI-based approach. However, details regarding the ground truth measurement, as well as the size of the experimental area, are missing, thus unfortunately making the analysis in [26] not suitable for application to UAVs.

In the context of radio localization, AoA estimation is attracting a significant interest, especially owing to the fact that, for some use cases, if combined with FTM distance estimation (as considered, for example, in [16]), it allows to estimate the position of a target with just one anchor. As drawback, AoA estimation requires higher computational complexity at the anchor, due to the need to integrate multiple antennas at specific distances and, thus, a larger number of RF-to-signal converters in the device modem: this increases the cost of the hardware equipment design and production. However, both these solutions (namely, FTM and AoA) are affected by the signal reflections due to the environment, which can significantly influence both the distance and angle estimation. According to the literature, a combination of AoA, channel state information (CSI) analysis, and ToF can be combined and exploited with a more complex matrix of antennas located on a UAV, not only allowing to achieve a high-level positioning accuracy, but also allowing to estimate pose and flight direction of the tracked UAV, as discussed in [27] and [28]. Although the results in [27] and [28] are quite impressive, several details regarding implementation and experimental setup are missing, thus making difficult to compare it with the position estimation framework proposed in this article, which provides a significant lower cost and complexity. However, the exploitation of the approaches discussed in [27] and [28], combined with FTM and more convenient hardware, might represent an interesting future research direction, as detailed in Section VII.

III. WI-FI-BASED LOCALIZATION

There exist several ways to exploit existing deployed Wi-Fi infrastructures for positioning purposes. In particular, two possible Wi-Fi localization approaches can be identified: 1) *passive* and 2) *active*. *Passive* localization techniques include Wi-Fi fingerprinting, as well as RSSI-based approaches. They are defined passive as the target does not need to exchange data with the existing infrastructure, but can only process the incoming signals transmitted by the APs, thus passively exploiting the available APs deployed in the environment. At the opposite, *active* localization techniques involve bidirectional information exchange between target and APs, as in a ToF-based approach.

A. Passive Localization

1) Wi-Fi Fingerprinting Localization: Wi-Fi fingerprinting localization exploits the Wi-Fi signals broadcasted by APs present in the localization environment. More in detail, it relies on the unique behavior of Wi-Fi signals upon interaction with the surrounding environment, which leads to variations in the signals' strengths and phases. In particular, the way Wi-Fi signals propagate in an indoor environment with respect to the target position can be considered similar to an RF "signature," which strictly depends on the environment and can be exploited for localization. This might happen by creating (through several *in-field* measurements) a heat map associated with a database containing RF data, such as APs' media access control (MAC) addresses and their RSSIs perceived by the target in different positions. The data in the heat map are then used as unique *fingerprints* for each measurement location. Obviously, this approach is effective if the environment does not change at all.

Once both the heat map and the corresponding measurements' database for the whole area of interest are available (through an operational phase denoted as *offline phase*), they can be used for the next operational phase (denoted as *online phase*), where the target device (to be localized) performs, at each position, a scan of the sensed Wi-Fi signals, thus obtaining relevant RF data, i.e., the radio fingerprint. Through the use of the gathered data, it can then search, within the RF database, the most likely location, i.e., the location with a recorded fingerprint most similar to the obtained fingerprint. Then, once (and *if*) a match is found, various algorithms can be used to estimate the device's location. On the basis of the measurement density (i.e., measurements per area unit) of the map built during the *offline phase*, as well as of the adopted algorithms (often relying on ML techniques), a position error of a few meters can be achieved [29]. While this technique is used on smartphones for indoor localization through shared heat maps [30], the accuracy of Wi-Fi fingerprinting depends on the signal variability and environmental changes. This implies that the system accuracy can be heavily affected by 1) environmental layout changes, 2) addition or removal of APs, as well as 3) interference incoming from other wireless devices.

2) Wi-Fi RSSI-Based Localization: A simpler approach, which does not require the need to a priori know the positions of the APs deployed in an environment, is based on the use of the RSSI measurements gathered from the APs. In this case, the RSSI, which is a measure of the received RF power obtained by the radio interface, is collected from the signal transmitted by each AP and is used to estimate the distance between the target (to be localized) and the AP. The distances from at least three APs need to be collected

for bidimensional localization purposes. More in detail, the following log-distance path loss model, in logarithmic scale, can be used [31]

$$P_i = P_0 + 10 \cdot \gamma \cdot \log_{10}\left(\frac{d}{d_0}\right) + X_g \qquad d \geq d_{\text{th}} \tag{1}$$

where P_i is the received signal power [dimension: (dBm)]; P_0 is the measured received signal power [dimension: (dBm)] at a reference distance d_0 [dimension: (m)]; γ is the path loss exponent (adimensional), a constant value typically between 2 and 4 depending on the environment; d_i is the distance between the AP and target node [dimension: (m)]; $d_{\text{th}} \geq 2 \cdot \lambda$ is the far-field region distance threshold, needed to make the log-distance path loss model valid [32]; and $X_g \sim \mathcal{N}(0, \sigma_g)$ is a normal distribution with zero mean, representing flat fading, used to model the signal envelope fluctuations and the corresponding gains in received power. A COTS device typically returns quantized values of the received signal power. For simplicity, in the rest of the mathematical derivations in this article, we will consider the RSSI corresponding to the received signal power—this corresponds to infinite quantization. In practice, the RSSI is a quantized version of the received signal power (the experimental data refer to the RSSI). Therefore, by measuring P_0, γ, and X_g in the target environment at d_0 (typically being $d_0 = 1$ m), it is possible to invert (1) to estimate d_i from the measured RSSI P_i as follows:

$$d_i = 10^{-\left(\frac{P_i - P_0 - X_g}{10 \cdot \gamma}\right)}. \tag{2}$$

Once at least three distances are estimated from different APs, it is possible to apply a multilateration algorithm and estimate the position of the target in the environment. However, while this approach has a very low implementation cost, since almost any Wi-Fi client can gather the RSSI from nearby APs, its accuracy is known to be low. This is especially true for medium- and long-range distances, where the log-distance path loss model, together with the fluctuations of the RSSI measurements, makes it very complicated to estimate the distance between the AP and the target, especially if the target is mobile. This is exacerbated by the fact that only a few RSSI samples can be collected (to make position estimation only real time), thus hindering to properly filter the environmental noise. In Fig. 1, we show the Wi-Fi RSSI data collected by a node at different distances in LOS conditions, showing how the measured data compare to the theoretical log-distance path loss model curve.

Moreover, RSSI-based localization also depends on the antennas used by the target and the APs, since they can significantly affect the RSSI values based on their gains and directivity. Therefore, a proper value of P_0 should be measured and defined for each device. Also, radio signal propagates in different ways according to the environment, since obstacles (such as walls, humans, and other material) can significantly hinder propagation, making distance estimation [based on the log-distance model in (1)] inaccurate.

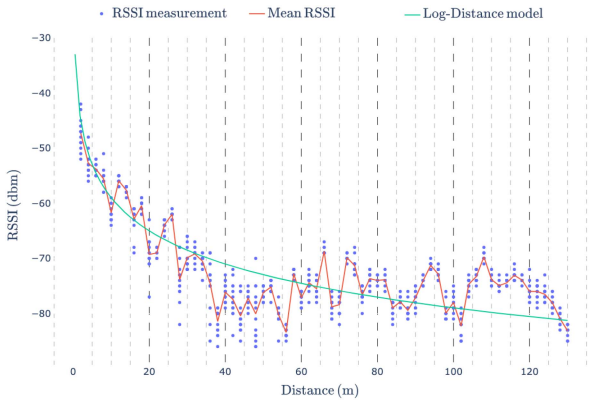


Fig. 1. Experimental measurement of the Wi-Fi RSSI at different distances: the blue dots are the measured values at different distances, the red line is the average RSSI value for each measurement point, and the green line is the theoretical log-distance path loss model curve.

B. Active Localization

1) Wi-Fi ToF-Based Localization: The most common active localization technique is based on ToF measurement, which has been officially introduced in the Wi-Fi IEEE 802.11mc amendment [17] standardized in June 2016, under the name of FTM [33]. Several COTS Wi-Fi chipsets already support the IEEE 802.11mc amendment, making meter-level localization feasible (at least in principle). More in detail, FTM utilizes round-trip time (RTT) measurements to determine distances between the Wi-Fi client and APs. FTM also exploits two-way ranging (TWR), thus not requiring any clock synchronization between the two parties, since the RTT is computed by exchanging several messages between the *initiator* (i.e., the Wi-Fi client) and the *responder* (i.e., the AP). However, unlike UWB technology, which, given the extremely short RF pulses (and ultra-large bandwidth), requires only one TWR frame exchange to estimate the RTT between two parties, Wi-Fi FTM requires to exchange several TWR frames in order to determine a reliable average value of the RTT and, therefore, the distance between the two parties. In Fig. 2, an illustrative FTM frame exchange sequence is shown. After the *initiator* performs a Wi-Fi scan of the surrounding environment, the process begins with the initiator sending an *FTM Request* frame to the target *responder*, which immediately replies with an *Acknowledgment (ACK)* frame to the *initiator*. At this point, according to the defined parameters (namely, the number of *FTM* frames to exchange), a sequence of *n FTM* and *ACK* frames, collectively defined as *Burst*, are exchanged between the two parties. More in detail, by setting $n = 4$, for the ith RTT with $i \in \{1, \ldots, 4\}$, these frames contain the time of departures (ToDs) $t_1^{(i)}$ and $t_3^{(i)}$, as well as the time of arrivals (ToAs) $t_2^{(i)}$ and $t_4^{(i)}$ of the previous exchanged frames. This allows the *responder* to compute RTT_i and send it back to the *initiator* as

$$\text{RTT}_i = \left(t_4^{(i)} - t_1^{(i)}\right) - \left(t_3^{(i)} - t_2^{(i)}\right) \tag{3}$$

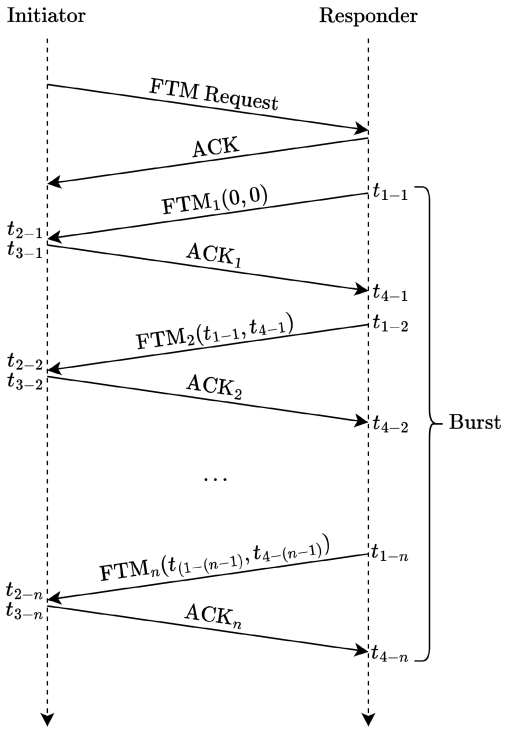


Fig. 2. FTM frame exchange sequence to determine the RTT between the *initiator* and the *responder*.

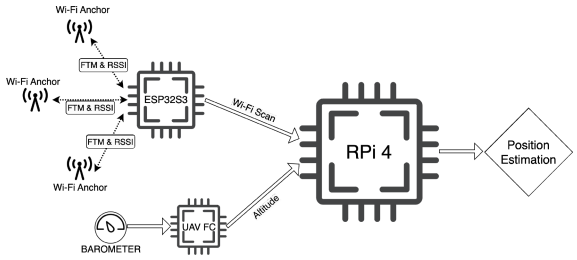


Fig. 3. High-level overview of the localization architecture components.

where $t_1^{(i)}$ is the ToD at which the *FTM* frame has been sent by the *responder*; $t_2^{(i)}$ is the ToA at which the *FTM* frame has been received by the *initiator*; $t_3^{(i)}$ is the ToD of the *ACK* frame; and $t_4^{(i)}$ is the ToA at which the ACK has been received by the *responder*. Since during a *Burst*, n RTT measurements $\{\text{RTT}_i\}_{i=1}^n$ are evaluated, the *initiator* computes the average RTT as

$$\overline{\text{RTT}} = \frac{1}{n} \cdot \sum_{i=1}^{n} \text{RTT}_i \,. \tag{4}$$

Therefore, it is finally possible to compute the ToF (denoted as τ_{tof}) as

$$\tau_{\text{tof}} = \frac{\overline{\text{RTT}}}{2} \,. \tag{5}$$

Consequently, the *initiator* can compute the distance d from the *responder* as

$$d = \tau_{\text{tof}} \cdot v + \epsilon_{\text{cal}} \tag{6}$$

where v is the speed of light [dimension: (m/s)], and ϵ_{cal} is an offset [dimension: (m)] used to calibrate the delays introduced by the specific (target) device. Once at least three distances from known APs are gathered, performing the multilateration allows to estimate the position of device.

According to the concepts of radio detection and ranging, the ranging accuracy—denoted as *range resolution* and defined as the system's ability to distinguish between two separate points (or objects) in space on the basis of their distances—is determined by the pulse duration and bandwidth of the radio signal. In detail, the range resolution (denoted as rr) can be estimated as [34]

$$rr \approx \frac{v}{2B} \tag{7}$$

where rr represents the attainable range resolution [dimension: (m)]; v denotes the velocity of the signal [dimension: (m/s)], which can be approximated to the speed of light (namely, approximately $3 \cdot 10^8$ m/s); and B is the system's bandwidth [dimension: (Hz)].

For a 2.4-GHz Wi-Fi system with $B = 40$ MHz, the smallest rr [that can be computed according to (7)] is approximately equal to 3.75 m, while it becomes equal to 7.5 m for $B = 20$ MHz 2.4 GHz Wi-Fi systems. Instead, in the case of a Wi-Fi system operating in the 5-GHz band with $B = 80$ MHz, it is possible to achieve $rr = 1.88$ m, while $rr = 0.94$ m can be reached with $B = 160$ MHz, as confirmed in [17].

Therefore, the advantages of the Wi-Fi FTM technique are mainly related to 1) its low implementation cost, since the most recent APs and devices already support this operational mode, and 2) its accuracy, higher than that of traditional RSSI-based localization solutions, since it is less affected by the presence of obstacles and signal attenuation and is not affected by the antennas' directivity. However, even this solution has some drawbacks: since the protocol is supported by different devices, a proper value of ϵ_{cal} must be taken into account for each specific device, limiting its interoperability only to well-known devices. Finally, unlike UWB, Wi-Fi FTM can be heavily affected by signal reflections in multipath scenarios.

In the following, both Wi-Fi RSSI and FTM will be evaluated for real-time UAV localization purposes: the obtained results will show that the FTM-based localization system outperforms the RSSI-based system when considering a UAV flying in a large operational area. In all cases, we adopt the cheapest COTS FTM hardware available on the market operating at 2.4 GHz with a 40-MHz bandwidth (namely, the ESP32S3 [35]).

C. Multilateration

Once the distance estimates between the target node and at least three anchors are gathered (through either passive or active approaches), it is possible to feed these data, together with the known positions of the APs, to a geometric multilateration algorithm, in charge of estimating the position of the

target device.[2] More in detail, since a multilateration algorithm [36] simply determines the intersection between multiple spheres centered in the (known) APs' position coordinates and with radii equal to the estimated distances, in order to estimate the device position in a bidimensional environment at least three APs are needed—in a 3-D space, instead, four APs are needed to identify a possible unique solution.

Given a list of known APs with coordinates (x_i, y_i, z_i), $i \in \{1, 2, \ldots, n\}$, together with the corresponding distances d_i, $i \in \{1, 2, \ldots, n\}$, between the device to be localized and the ith AP, estimated according to one of the methods detailed and discussed in Sections III-A and III-B, it is possible to analytically define, at each measurement point, a system of equations, where each equation represents the squared distance of the target from an AP

$$\begin{cases} (P_x - x_1)^2 + (P_y - y_1)^2 + (P_z - z_1)^2 = d_1^2 \\ (P_x - x_2)^2 + (P_y - y_2)^2 + (P_z - z_2)^2 = d_2^2 \\ \quad \vdots \\ (P_x - x_n)^2 + (P_y - y_n)^2 + (P_z - z_n)^2 = d_n^2 \end{cases} \tag{8}$$

where (P_x, P_y, P_z) represent the coordinates of the position of the device to be localized in the 3-D space.[3]

The system of equations in (8) can be solved by adopting an optimization algorithm. More in detail, in this work the least squares (LS) optimization algorithm [37] has been adopted to find the best fitting coordinates of the point P, which minimize the sum of squared residuals of (8). Hence, it is possible to define the following objective function from system (8):

$$C_{(P_x,P_y,P_z)} = \sum_{i=1}^{n} \left[(P_x - x_i)^2 + (P_y - y_i)^2 + (P_z - z_i)^2 - d_i^2 \right]^2 . \tag{9}$$

The goal of an LS approach is to find the best fitting coordinates (P_x, P_y, P_z), minimizing the residuals of the objective function (9). More in detail, the residual r_i, $i \in \{1, 2, \ldots, n\}$, corresponds to the difference between the left-hand side and right-hand side of the ith equation in (8)

$$r_i = (P_x - x_i)^2 + (P_y - y_i)^2 + (P_z - z_i)^2 - d_i^2 . \tag{10}$$

Therefore, (9) can be expressed, in terms of residuals, as

$$C_{(P_x,P_y,P_z)} = \sum_{i=1}^{n} r_i^2 . \tag{11}$$

In order to minimize $C_{(P_x,P_y,P_z)}$, it is necessary to find its gradient with respect to the variables (P_x, P_y, P_z). Recall that the gradient is the following vector of partial derivatives:

$$\nabla C_{(P_x,P_y,P_z)} = \left[\frac{\partial C_{(P_x,P_y,P_z)}}{\partial P_x}, \frac{\partial C_{(P_x,P_y,P_z)}}{\partial P_y}, \frac{\partial C_{(P_x,P_y,P_z)}}{\partial P_z} \right] \tag{12}$$

where

$$\begin{aligned} \frac{\partial C_{(P_x,P_y,P_z)}}{\partial P_x} &= 2 \sum_{i=1}^{n} r_i \frac{\partial r_i}{\partial P_x} \\ \frac{\partial C_{(P_x,P_y,P_z)}}{\partial P_y} &= 2 \sum_{i=1}^{n} r_i \frac{\partial r_i}{\partial P_y} \\ \frac{\partial C_{(P_x,P_y,P_z)}}{\partial P_z} &= 2 \sum_{i=1}^{n} r_i \frac{\partial r_i}{\partial P_z} . \end{aligned} \tag{13}$$

Since

$$\begin{aligned} \frac{\partial r_i}{\partial P_x} &= 2(P_x - x_i) \\ \frac{\partial r_i}{\partial P_y} &= 2(P_y - y_i) \\ \frac{\partial r_i}{\partial P_z} &= 2(P_z - z_i) \end{aligned} \tag{14}$$

by combining (13) and (14), it is possible to obtain the following expression for the gradient of the objective function given by (11):

$$\nabla C_{(P_x,P_y,P_z)} = \left[2 \sum_{i=1}^{n} r_i \cdot 2(P_x - x_i), \times 2 \sum_{i=1}^{n} r_i \cdot 2(P_y - y_i), 2 \sum_{i=1}^{n} r_i \cdot 2(P_z - z_i) \right] . \tag{15}$$

Finally, in order to find the optimal solution (under the assumption that the objective function is convex), the gradient must be set to zero

$$\nabla C_{(P_x,P_y,P_z)} = [0, 0, 0] . \tag{16}$$

The optimal solution can be found, for example, by adopting the least squares with multiple right-hand sides (LSMR) [38] algorithm, that iteratively updates the values of (P_x, P_y, P_z) using the gradient of the cost function and the learning rate α (to be properly optimized) to minimize the objective function. More precisely, at step $m \geq 1$, the updated values of (P_x, P_y, P_z), denoted as $(P_x^{(m)}, P_y^{(m)}, P_z^{(m)})$, can be expressed as

$$\begin{aligned} P_x^{(m)} &= P_x^{(m-1)} - \alpha \cdot \nabla C_{(P_x,P_y,P_z)} \\ P_y^{(m)} &= P_y^{(m-1)} - \alpha \cdot \nabla C_{(P_x,P_y,P_z)} \\ P_z^{(m)} &= P_z^{(m-1)} - \alpha \cdot \nabla C_{(P_x,P_y,P_z)} \end{aligned} \tag{17}$$

where $(P_x^{(m-1)}, P_y^{(m-1)}, P_z^{(m-1)})$ are the values at the previous step. The starting values used as initial guess—namely, $(P_x^{(0)}, P_y^{(0)}, P_z^{(0)})$—are set to $(1, 1, 1)$. This iterative optimization process continues until the cost function converges to a minimum. Denoting the final iteration step as m_{fin}, the optimal position estimate is given by $(P_x^{(m_{\text{fin}})}, P_y^{(m_{\text{fin}})}, P_z^{(m_{\text{fin}})})$.

In order to further optimize the process to find the best fitting coordinates, it is possible to insert constraints into the LS optimization problem, to ensure that the estimated parameters lie within a specific range or satisfy certain conditions. For instance, in Wi-Fi-based UAV localization,

[2]Other nongeometric algorithms could be applied. However, this research direction is not investigated to keep the computational complexity low.

[3]For simplicity, the target node is associated with a single point, ideally corresponding to its center of mass.

it might be useful to apply a constraint to P_z, as it cannot be negative. Moreover, it is also possible to rely on the UAV's *on-board* barometer to estimate the flight altitude (i.e., P_z), which is already computed and provided by the UAV's flight controller (FC). Therefore, P_z can be assumed to be known, thus simplifying (13) as

$$\begin{aligned}\frac{\partial C_{(P_x,P_y)}}{\partial P_x} &= 2\sum_{i=1}^{n} r_i \frac{\partial r_i}{\partial P_x} \\ \frac{\partial C_{(P_x,P_y)}}{\partial P_y} &= 2\sum_{i=1}^{n} r_i \frac{\partial r_i}{\partial P_y}\end{aligned} \tag{18}$$

where

$$\begin{aligned}\frac{\partial r_i}{\partial P_x} &= 2(P_x - x_i) \\ \frac{\partial r_i}{\partial P_y} &= 2(P_y - y_i)\,.\end{aligned} \tag{19}$$

This leads to the following simplified expression of the gradient (15) of the objective function:

$$\nabla C_{(P_x,P_y)} = \left[2\sum_{i=1}^{n} r_i \cdot 2(P_x - x_i), 2\sum_{i=1}^{n} r_i \cdot 2(P_y - y_i)\right]. \tag{20}$$

IV. KALMAN FILTERING

In order to enhance the Wi-Fi-based position estimation accuracy, it could be useful (and might be considered a good practice) to properly filter the distances estimated according to the RSSI- or FTM-based methods described in Section III, in order to remake outliers (caused by the radio environmental noise affecting the received Wi-Fi signals at the device to be localized). Among various possible filtering approaches, the most common and adopted solution is based on the KF [39], a recursive filter adopted in heterogeneous contexts (e.g., control systems, signal processing, localization, finance, etc.) to estimate and predict the state of linear and nonlinear systems with noisy measurement inputs.

In detail, Kalman Filters (KFs) have been initially proposed for linear systems, since they require low computational capabilities with respect to more complex approaches. However, in the last years, EKFs [40] have been proposed for nonlinear systems, also thanks to the increased computational capabilities of mobile devices. More in detail, both KF and EKF iteratively combine information from previous estimates and current measurements to provide an accurate (and, in some cases, optimal) estimate of the state of a system, even in the presence of noisy measurements and uncertainty, thus allowing to significantly reduce the impact of measurement errors on the final accuracy.

When dealing with a KF, system dynamics are crucial for the filtering process, eventually determining the overall filter performance. Therefore, it is necessary to define the transition and observation models that describe how the state of the dynamic system evolves over time and how it relates to measurements, respectively.

The *state transition model* of a KF can be generally expressed as [41]

$$\mathbf{x}_t = \mathbf{F}_t \cdot \mathbf{x}_{t-1} + \mathbf{B}_t \cdot \mathbf{u}_t + \mathbf{w}_t \tag{21}$$

where $\mathbf{w}_t \sim \mathcal{N}(0, \mathbf{Q}_t)$ represents the process noise, characterized by a normal distribution with zero mean and covariance matrix $\mathbf{Q}_t$; $\mathbf{x}_t$ and $\mathbf{x}_{t-1}$ are the current and previous estimated system state vectors at time t and $t-1$, respectively; $\mathbf{F}_t$ is the state transition matrix at time t; and $\mathbf{u}_t$ is the control input vector, while $\mathbf{B}_t$ is the control input matrix, both at time t.

The *observation model* can be expressed as [41]

$$\mathbf{z}_t = \mathbf{H}_t \cdot \mathbf{x}_t + \mathbf{v}_t \tag{22}$$

where $\mathbf{v}_t \sim \mathcal{N}(0, \mathbf{R}_t)$ is the observation noise, characterized by a normal distribution with zero mean and covariance matrix $\mathbf{R}_t$; $\mathbf{z}_t$ is the observation (or measurement) vector; and $\mathbf{H}_t$ is the observation matrix that describes the relationship between the state information vector $\mathbf{x}_t$ and the observation vector $\mathbf{z}_t$.

In order to properly estimate the new state of the system, the KF relies on two phases [42]: 1) *state predict* and 2) *state update*. In particular, in the *state predict* phase, the new state is predicted using the state transition model (21), while the *state update* phase refines the state estimate by incorporating the measurements using the observation model (22). This iterative process allows the KF to estimate the state of a linear dynamic system in the presence of noise and measurement uncertainties.

The *state predict* phase, which predicts the a priori state information vector, denoted as $\hat{\mathbf{x}}_{t|t-1}$, using the information from the previous a posteriori state information vector, denoted as $\mathbf{x}_{t-1|t-1}$, relies on the following equation [41]:

$$\hat{\mathbf{x}}_{t|t-1} = \mathbf{F}_t \cdot \mathbf{x}_{t-1|t-1} + \mathbf{B}_t \cdot \mathbf{u}_t \tag{23}$$

where $\mathbf{u}_t$ is the system control input vector; $\mathbf{F}_t$ is the state transition matrix; and $\mathbf{B}_t$ is the control input matrix.

The estimated a priori covariance matrix $\hat{\mathbf{P}}_{t|t-1}$ is obtained as [41]

$$\hat{\mathbf{P}}_{t|t-1} = \mathbf{F}_t \cdot \mathbf{P}_{t-1|t-1} \cdot \mathbf{F}^{\mathsf{T}} + \mathbf{Q}_t \tag{24}$$

where $\mathbf{P}_{t-1|t-1}$ is the previous state a posteriori covariance matrix, and $\mathbf{Q}_t$ denotes the covariance matrix of the noise process.

The *State Update* phase corrects the a priori state information through the use of the observation vector. In detail, in this phase, the innovation vector $\tilde{\mathbf{y}}_t$, representing the difference between the observed measurement and the predicted measurement based on the current state estimate, can be expressed as [41]

$$\tilde{\mathbf{y}}_t = \mathbf{z}_t - \mathbf{H}_t \cdot \hat{\mathbf{x}}_{t|t-1}\,. \tag{25}$$

On the other hand, the covariance matrix $\mathbf{S}_t$ can be expressed as [41]

$$\mathbf{S}_t = \mathbf{H}_t \cdot \hat{\mathbf{P}}_{t|t-1} \cdot \mathbf{H}_t^{\mathsf{T}} + \mathbf{R}_t\,. \tag{26}$$

At this point, the updated a posteriori state vector $\hat{\mathbf{x}}_{t|t}$ can be calculated as [41]

$$\hat{\mathbf{x}}_{t|t} = \hat{\mathbf{x}}_{t|t-1} + \mathbf{K}_t \cdot \widetilde{\mathbf{y}}_t \tag{27}$$

where $\mathbf{K}_t$ is the *optimal* Kalman gain, which can be obtained as follows [41]:

$$\mathbf{K}_t = \hat{\mathbf{P}}_{t|t-1} \cdot \mathbf{H}_t^{\mathsf{T}} \cdot \mathbf{S}_t^{-1} . \tag{28}$$

Finally, the updated a posteriori state covariance matrix $\mathbf{P}_{t|t}$ can be derived as [41]

$$\mathbf{P}_{t|t} = (\mathbf{I} - \mathbf{K}_t \cdot \mathbf{H}_t) \cdot \hat{\mathbf{P}}_{t|t-1} . \tag{29}$$

Once the KF has been set, the aforementioned parameters (namely, $\mathbf{P}_0$, $\mathbf{R}_t$, $\mathbf{Q}_t$, $\mathbf{H}_t$, $\mathbf{u}_t$, and $\mathbf{F}_t$) must be properly set according to the intended filtering application. In the developed Wi-Fi-based localization system (detailed in Section V), the KFs are applied to two different parts of the system, namely: 1) at the reception of the RSSI or FTM measurement from each AP, and 2) to further filter the estimated RSSI- or FTM-based positions computed by the multilateration algorithm described in Section III-C. Moreover, different KFs properly tuned with the right parameters must be defined accordingly to efficiently filter the measurement noise. The reason behind the double KF implementation in the developed system is detailed in the following.

The *first* KF, denoted as "prefilter RSSI-KF," is applied to the RSSI measurement arriving from the Wi-Fi scanner of the device to be localized, with the aim to remove possible measurement outliers. More in detail, in the case of the RSSI-based localization system, an RSSI-KF$_{\text{AP}_i}$ is assigned to each known AP$_i$, $i \in \{1, \ldots, 6\}$, thus allowing to filter the measured RSSI according to the previous measurement of the same AP$_i$. This means that, in the experimental setup of the developed system with six APs, six RSSI-KF$_{\text{AP}_i}$ (one for each AP) are initialized by the developed localization algorithm. Therefore, in the RSSI-KF$_{\text{AP}_i}$, the estimated 1×1 vector $\mathbf{x}_t^{(\text{AP}_i)}$ given by (21) is defined as

$$\mathbf{x}_t^{(\text{AP}_i)} = \hat{\text{RSSI}}_t^{(\text{AP}_i)} \tag{30}$$

while the observed 1×1 vector $\mathbf{z}_t^{(\text{AP}_i)}$ of (22) is defined as

$$\mathbf{z}_t^{(\text{AP}_i)} = \text{RSSI}_t^{(\text{AP}_i)} . \tag{31}$$

Since the estimated vectors have only one element, all the aforementioned matrices have one element. Moreover, all $\{\text{RSSI-KF}_{\text{AP}_i}\}_{i=1}^{6}$ share the same initial covariance 1×1 matrix $\mathbf{P}_0 = 1$, the same observation noise 1×1 matrix $\mathbf{R}_t = 3 \cdot \mathbf{I} = 3$, and the same process noise 1×1 matrix $\mathbf{Q}_t = 0.2 \cdot \mathbf{I} = 0.2$. Those values have been chosen according to *in-field* experimental measurements of the RSSI in both static and mobile conditions of the device to be localized, as further detailed in Section VI. When the RSSI-KF$_{\text{AP}_i}$ is initialized, the initial estimated states $\mathbf{x}_0^{(\text{AP}_i)}$ are set equal to the first observations $\mathbf{z}_0^{(\text{AP}_i)}$. Since in the developed system there is no control input signal $\mathbf{u}_t$, this term can be neglected, thus also removing $\mathbf{B}_t$. Instead, $\mathbf{H}_t$ and $\mathbf{F}_t$ are equal to the identity 1×1 matrix $\mathbf{I} = \mathbf{1} = 1$.

The same approach (i.e., initializing a filter for each AP$_i$, $i \in \{1, \ldots, 6\}$) is applied to the FTM. In detail, an FTM-KF$_{\text{AP}_i}$ is assigned to each known AP$_i$, thus allowing to filter the measured FTM according to the previous measurement of the same AP$_i$. In the FTM-KF$_{\text{AP}_i}$, the estimated 1×1 vector $\mathbf{x}_t^{(\text{AP}_i)}$ is defined as

$$\mathbf{x}_t^{(\text{AP}_i)} = \hat{\text{FTM}}_t^{(\text{AP}_i)} \tag{32}$$

while the observed 1×1 vector $\mathbf{z}_t^{(\text{AP}_i)}$ can be expressed as

$$\mathbf{z}_t^{(\text{AP}_i)} = \text{FTM}_t^{(\text{AP}_i)} . \tag{33}$$

All $\{\text{FTM-KF}_{\text{AP}_i}\}_{i=1}^{6}$ share the same initial covariance 1×1 matrix $\mathbf{P}_0 = 1$, the same observation noise 1×1 matrix $\mathbf{R}_t = 2 \cdot \mathbf{I} = 2$, and the same process noise 1×1 matrix $\mathbf{Q}_t = 0.2 \cdot \mathbf{I} = 0.2$. Similarly to the RSSI-based case, those values have been chosen according to *in-field* experimental measurements of the FTM in both static and mobile conditions of the device to be localized—this will be further detailed in Section VI. Then, as for RSSI-KF filters, the initial estimated state $\mathbf{x}_0^{(\text{AP}_i)}$ is set equal to the first observation $\mathbf{z}_0^{(\text{AP}_i)}$ to initialize the corresponding filter. As for RSSI-KF, in this case as well the terms $\mathbf{u}_t$ and $\mathbf{B}_t$ are neglected, while $\mathbf{H}_t$ and $\mathbf{F}_t$ are set to the identity 1×1 matrix $\mathbf{I} = \mathbf{1} = 1$.

The *second* set of KFs, corresponding to the multilateration filters and denoted as POS-KFs, are applied to the estimated RSSI-based and FTM-based positions—namely, $(P_x^{(\text{RSSI})}, P_y^{(\text{RSSI})}, P_z^{(\text{BARO})})$ and $(P_x^{(\text{FTM})}, P_y^{(\text{FTM})}, P_z^{(\text{BARO})})$, respectively—computed by the multilateration algorithm applied to RSSI and FTM data, respectively. Therefore, the estimated state 3×1 vectors $\mathbf{x}_t^{(\text{RSSI})}$ of the POS-KF$^{(\text{RSSI})}$ applied to the RSSI-based position are defined as

$$\mathbf{x}_t^{(\text{RSSI})} = \left(\hat{P}_x^{(\text{RSSI})}, \hat{P}_y^{(\text{RSSI})}, \hat{P}_z^{(\text{BARO})}\right)_t^{\mathsf{T}} \tag{34}$$

while the observed 3×1 vectors $\mathbf{z}_t^{(\text{RSSI})}$ are defined as

$$\mathbf{z}_t^{(\text{RSSI})} = \left(P_x^{(\text{RSSI})}, P_y^{(\text{RSSI})}, P_z^{(\text{BARO})}\right)_t^{\mathsf{T}} . \tag{35}$$

The estimated state 3×1 vectors $\mathbf{x}_t^{(\text{FTM})}$ of the POS-KF$^{(\text{FTM})}$ applied to the FTM-based position are defined as

$$\mathbf{x}_t^{(\text{FTM})} = \left(\hat{P}_x^{(\text{FTM})}, \hat{P}_y^{(\text{FTM})}, \hat{P}_z^{(\text{BARO})}\right)_t^{\mathsf{T}} \tag{36}$$

while the observed 3×1 vectors $\mathbf{z}_t^{(\text{FTM})}$ can be expressed as

$$\mathbf{z}_t^{(\text{FTM})} = \left(P_x^{(\text{FTM})}, P_y^{(\text{FTM})}, P_z^{(\text{BARO})}\right)_t^{\mathsf{T}} . \tag{37}$$

In both cases, the estimated vector has dimensions 3×1 and, therefore, all matrices have dimensions 3×3. The initial covariance matrix $\mathbf{P}_0$ 3×3 matrix is defined as

$$\mathbf{P}_0 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{38}$$

while the 3×3 observation noise covariance matrix $\mathbf{R}_t$ is defined as

$$\mathbf{R}_t = \begin{bmatrix} 0.8 & 0 & 0 \\ 0 & 0.8 & 0 \\ 0 & 0 & 0.8 \end{bmatrix} \tag{39}$$

(a) (b)

Fig. 4. Equipment used for the Wi-Fi localization test. (a) Quadcopter with the on-board ESP32S3 module connected to the RPi4, and (b) one of the six ESP32S3 APs, held by a prototypical 3-D-printed case, deployed in the experimental environment.

and the 3×3 process noise covariance matrix $\mathbf{Q}_t$ as

$$\mathbf{Q}_t = \begin{bmatrix} 0.2 & 0 & 0 \\ 0 & 0.2 & 0 \\ 0 & 0 & 0.2 \end{bmatrix}. \tag{40}$$

Hence, when the filters are initialized, the initial estimated states $\mathbf{x}_0^{(\text{RSSI})}$ and $\mathbf{x}_0^{(\text{FTM})}$ are set equal to the first observations $\mathbf{z}_0^{(\text{RSSI})}$ and $\mathbf{z}_0^{(\text{FTM})}$, respectively. Since there is no control input signal, the terms $\mathbf{u}_t$ and $\mathbf{B}_t$ are neglected, while $\mathbf{H}_t$ and $\mathbf{F}_t$ are set equal to the 3×3 identity matrix $\mathbf{I}$.

All the aforementioned KFs' parameters have been chosen according to experimental evaluations carried out in both static and mobile conditions, aiming at finding an effective KF tuning process, representing an acceptable *tradeoff* for both scenarios, as will be further discussed in the following. This allows to understand when the use of both filters can be beneficial, as well as when the use of only one of the identified KFs (the *first* or the *second*) is preferable.

V. SYSTEM ARCHITECTURE

The developed low-cost real-time Wi-Fi-based localization system is composed of several elements. On the hardware side, the Wi-Fi RSSI and FTM measurements are gathered on a common platform (used for both the APs and the target device to be localized) based on the ESP32S3 SoC, embedded in the Lilygo T3S3 development board [43]. Then, one Lilygo T3S3 has been integrated (inside its own 3D-printed case) on the Tarot 650-based UAV platform, as shown in Fig. 4(a), in turn connected to a Raspberry Pi 4 (RPi4) single board computer located *on-board* the drone and running Ubuntu 20.04 OS, as well as the robot operating system (ROS) framework [44]. A high-level representation of the proposed experimental localization system is shown in Fig. 3. Our goal is to gather ground truth satellite data (from the GNSS receiver attached to the UAV's FC) as well as the Wi-Fi positioning data of the designed system, thus allowing to properly evaluate, in a comparative way with respect to the only GNSS-based position, its performance. Six Lilygos T3S3, together with their 3D-printed cases [as shown in Fig. 4(b)], have been adopted as Wi-Fi APs and deployed in the experimental environment.

The ESP32S3 firmware has been customized according to the role of the Lilygo T3S3.

1) In the boards used as Wi-Fi APs, the developed firmware has been properly designed to enable the ESP32S3 SoC to work as a Wi-Fi FTM *responder* on the 2.4-GHz band (with $B = 40$ MHz).
2) The development board integrated on the UAV, connected to the *on-board* RPi4 through a USB-C cable, has been programmed to act as a Wi-Fi node continuously scanning for available Wi-Fi APs.

The goal of the UAV is to retrieve the MAC address, the Service Set IDentifier (SSID), and the RSSI (with 1 dBm granularity) of each Wi-Fi AP, in turn performing as an FTM *initiator* with the FTM-enabled APs.

With regard to the FTM operational steps (detailed in Section III-B), in order to reduce the time needed to complete the computation of $\overline{\text{RTT}}$, the number of *FTM* frames to be exchanged for each *Burst* has been set equal to 8, with a total of only one *Burst* exchanged: this is expedient to minimize the impact of the movement of the quadcopter. In the end, this leads to a time duration of the *Burst* equal to 32 ms: considering the Wi-Fi scan time and the six APs used for the localization experiment, this results, in turn, in a measured average iteration period (denoted as θ_{cycle}) equal to 522 ms. Then, for the sake of clarity and readability, details on the Wi-Fi scanning data retrieval are discussed in Appendix A.

At the UAV side, the RPi4 receives these JavaScript Object Notation (JSON)-formatted strings and, through a Python script, separates the known APs—the ones used for the localization task, whose GNSS coordinates are known—from those corresponding to unknown APs. Once the known APs' list is built, RSSI-KF$_{\text{AP}_i}$, $i \in \{1, 2, \ldots, 6\}$, is set to the newly observed value RSSI$_{\text{AP}_i}$. The same operation is carried out with FTM-KF$_{\text{AP}_i}$, $i \in \{1, 2, \ldots, 6\}$, which is instead set to the measured value FTM$_{\text{AP}_i}$. At system activation as well as when a new AP is added to the list, the relative KFs are initialized as described in Section IV. Once the input data are filtered: in case of RSSI-based localization, the log-distance path loss model is applied to the filtered RSSI$_{\text{AP}_i}^{(\text{FILT})}$; while for the FTM-based solution, the calibration offset ϵ_{cal} is removed from the filtered FTM$_{\text{AP}_i}^{(\text{FILT})}$ measurement. In detail, for the log-distance path loss model behind the RSSI, $\gamma = 3.65$ and $P_0 = -27$ dBm have been measured in the experimental environment at the reference distance $d_0 = 1$ m; for the FTM calibration, according to the experimental measurements, the offset calibration has been set to $\epsilon_{\text{cal}} = -52.82$ m.

The output of log-distance estimation and FTM offset removal blocks, namely, $d_{\text{AP}_i}^{(\text{RSSI})}$ and $d_{\text{AP}_i}^{(\text{FTM})}$, then feed the multilateration algorithm, together with the altitude [dimension: (m)], which corresponds to $P_z^{(\text{BARO})}$ of the device to be localized and is retrieved from the UAV's FC (which estimates it through an internal barometer). Before applying the multilateration algorithm, the GNSS coordinates of

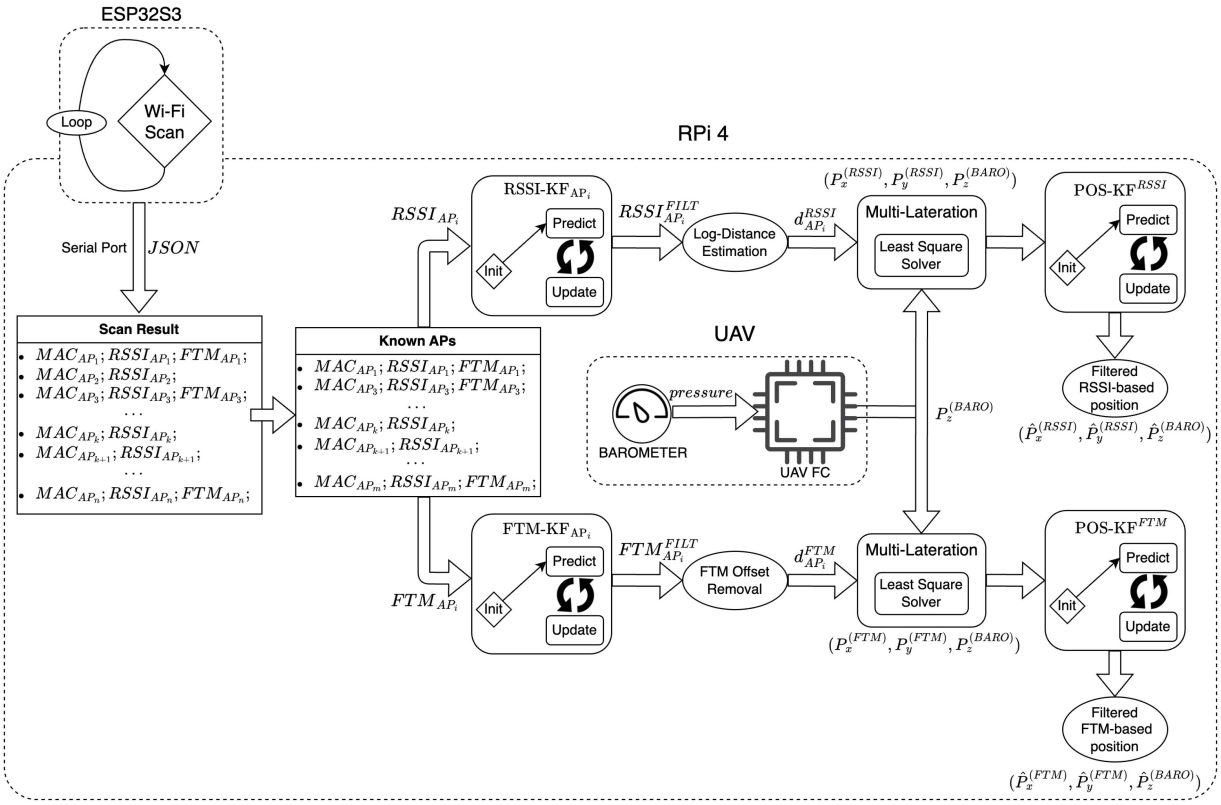


Fig. 5. System architecture of the developed Wi-Fi-based localization solution.

the known APs are converted from the World Geodetic System 1984 (WGS84) [45] standard to Earth-centered Earth-fixed (ECEF) [46] coordinates: this allows to apply the multilateration algorithm together with the LS. Once both RSSI-based ($P_x^{(\mathrm{RSSI})}$, $P_y^{(\mathrm{RSSI})}$, $P_z^{(\mathrm{BARO})}$) and FTM-based ($P_x^{(\mathrm{FTM})}$, $P_y^{(\mathrm{FTM})}$, $P_z^{(\mathrm{BARO})}$) coordinates have been computed, they are fed into the POS-KF filters, namely, one for the FTM-based localization and one for the RSSI-based localization. Finally, the output of the latter KFs, namely, the filtered RSSI-based estimated position coordinates ($\hat{P}_x^{(\mathrm{RSSI})}$, $\hat{P}_y^{(\mathrm{RSSI})}$, $\hat{P}_z^{(\mathrm{BARO})}$) and the filtered FTM-based estimated position coordinates ($\hat{P}_x^{(\mathrm{FTM})}$, $\hat{P}_y^{(\mathrm{FTM})}$, $\hat{P}_z^{(\mathrm{BARO})}$), are converted from ECEF to WGS84 coordinates and become the final estimated positions (according to the RSSI or FTM) of the UAV. A block diagram of the overall proposed system architecture is shown in Fig. 5.

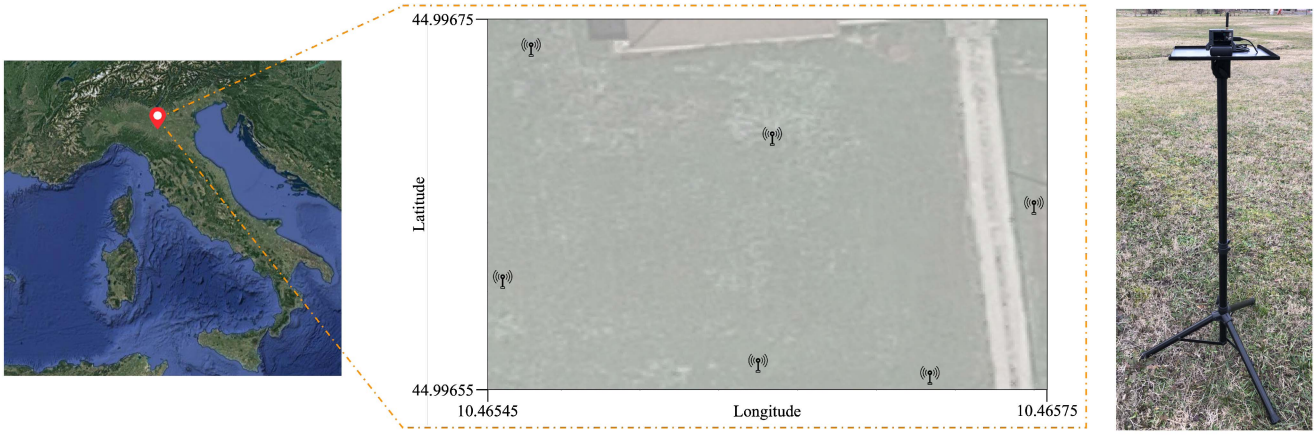
Finally, in order to ease performance evaluation (which will be the focus of Section VI), all *raw* Wi-Fi scan data, as well as the computed RSSI and FTM coordinates, are published to a custom-made ROS topic, thus allowing to record the estimated positions to be synchronized with the ground truth positions output by the UAV's GNSS receiver.

VI. EXPERIMENTAL PERFORMANCE EVALUATION

In order to evaluate the performance of the proposed Wi-Fi-based localization system, the considered testbed, as shown in Fig. 6(a), is located in an outdoor yard (measuring 26×21 m^2) in a suburban area in northern Italy—in the neighborhood of the city of Sabbioneta, Italy. More in detail, the battery-powered APs have been deployed on several tripods [shown in Fig. 6(b)] at the same altitude (equal to 150 cm) along the external border of the yard. The GNSS coordinates (in terms of latitude and longitude) of the APs have been obtained by averaging the positions gathered using the GNSS-receiver of the drone positioned on the tripod for approximately 5 min—averaging allows to minimize the measurable drift affecting GNSS systems. In detail, the point cloud composed by the GNSS measurements collected during 5 min at each AP lies in a circle with a radius of approximately 1 m: the reference coordinates of the AP to be used for the multilateration algorithm correspond the average values of the latitudes and longitudes of all points in the cloud.

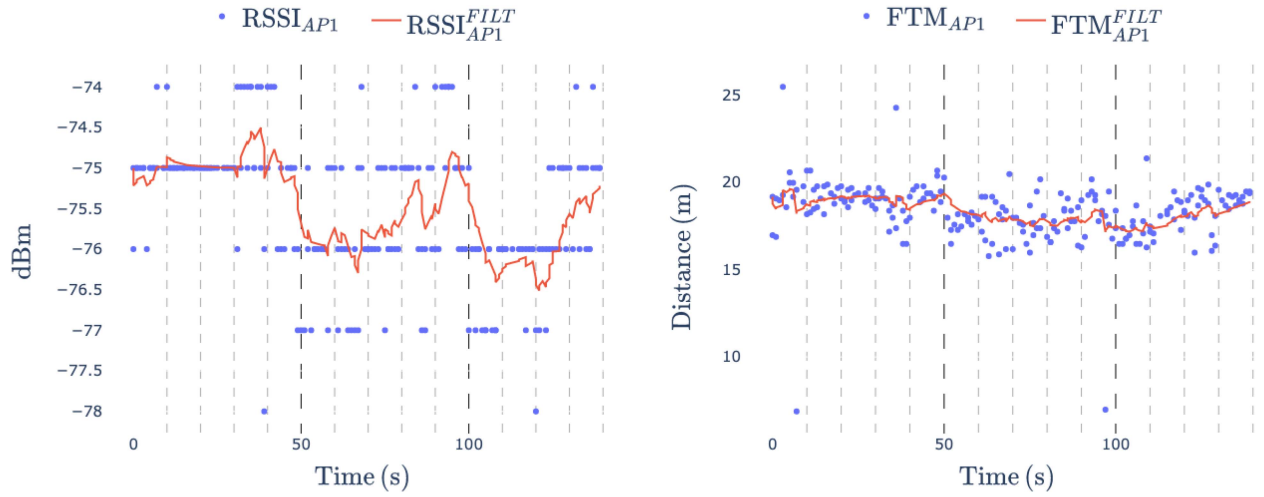
All APs have been deployed in LOS conditions with each other and with respect to the UAV. Given the suburban environment, a few Wi-Fi 2.4 GHz networks are available in the same area: in order to mitigate the possible interference effects, the deployed Wi-Fi APs have been set to operate in unused Wi-Fi channels.

The experimental evaluation has been carried out according to two procedures: 1) a *static* localization test, where the UAV hovered for about 2 min in the same position at approximately 2 m above ground level (AGL), with just a few fluctuations due to the correction applied by the FC to compensate wind and altitude drift; 2) a *mobile* localization



(a) (b)

Fig. 6. (a) Experimental testbed located in a yard in a suburban area in the neighborhood of the city of Sabbioneta, Italy, in the northern Italy. (b) Detail of a battery-powered AP deployed on a tripod.



(a) (b)

Fig. 7. Evaluation of the implemented KFs in static hovering conditions. (a) Unfiltered and filtered AP_1 RSSI measurements, and (b) unfiltered and filtered AP_1 FTM distance measurements.

test, where the UAV flew in AUTO mode at 2 m AGL with an average horizontal speed of 1.3 m/s for three times through a predefined flight path with a square shape, with a size of approximately 16×11 m^2, within the experimental area surrounded by the APs.

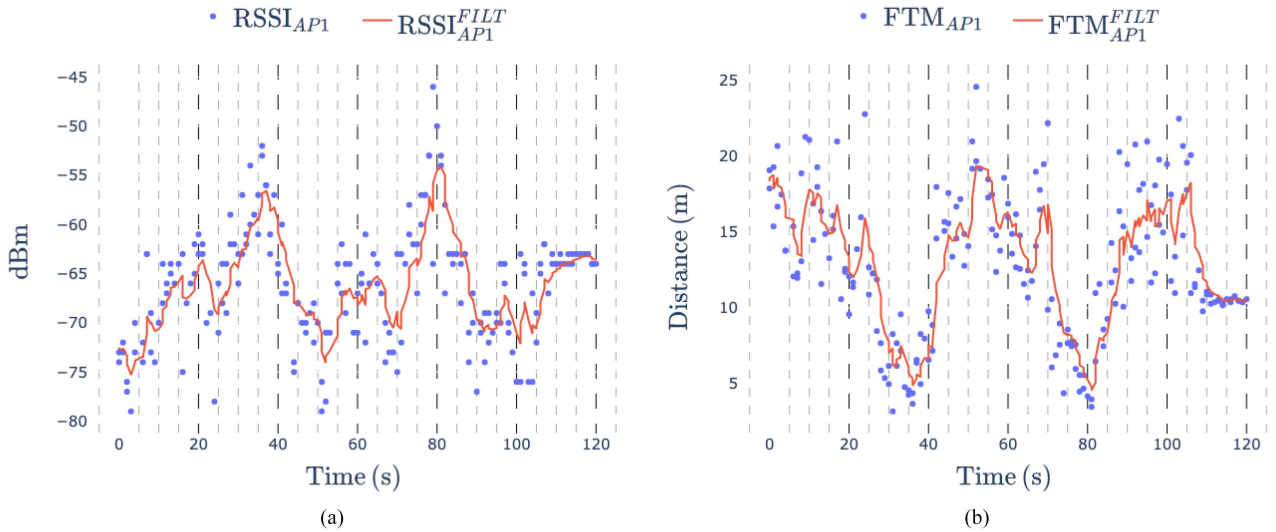
These tests have been carried out aiming at validating both RSSI- and FTM-based localization performance, in static (first procedure) and mobile (second procedure) conditions, in order to understand how accurate the proposed system can be in terms of real-time tracking of an object.

A. KFs Evaluation

Before performing the *in-flight* tests, we have carried out a preliminary performance evaluation of the KFs implemented and adopted to mitigate the noise of the RSSI and FTM measurements, aiming at properly setting their parameters outlined in Section IV.

In the *static hovering experimental evaluation*, carried out with respect to AP_1 (whose results are shown in Fig. 7), the KF performs outlier removal, filtering the measurement noise due to both environmental background noise and measurement error from the Wi-Fi scanner. In Fig. 7(a), the unfiltered and filtered RSSI measurements are shown, highlighting how the filter manages to remove some RSSI measurements at the boundaries of the observed RSSI's range, thus smoothing the RSSI values used for the distance estimation. In Fig. 7(b), the unfiltered and filtered FTM-based distance measurements are shown: as in the previous case, the KF manages to efficiently remove some FTM outliers, thus smoothing the FTM distance estimation.

Under UAV *mobile conditions* along the predefined path, the filtering performance is evaluated in Fig. 8. According to the results shown in Fig. 8(a), the KF manages to properly remove most of the outliers and to smooth the RSSI values. The same holds true for the KF applied to the



(a) (b)

Fig. 8. Evaluation of the implemented KFs with UAV in mobile condition. (a) Unfiltered and filtered AP_1 RSSI measurements, and (b) unfiltered and filtered AP_1 FTM measurements.

FTM measurements, with results shown in Fig. 8(b). This confirms the effectiveness of the designed filter, suitable to remove some background noise without losing relevant information, as would happen with other filters (e.g., exponential moving average [47])—the results with other filters are not shown here for the sake of conciseness.

B. Absolute Positioning Error Evaluation

In the following, the localization accuracy of the proposed system has been evaluated in the two aforementioned scenarios (namely, with static hovering UAV and with mobile UAV). However, in order to evaluate the impact of the implemented KFs and to understand how much and when it is beneficial to use both, the localization performance (in terms of positioning error with respect to the GNSS ground truth) is investigated in four different system setups for both RSSI- and FTM-based localization solutions depicted in Fig. 9 and illustrated in the following.

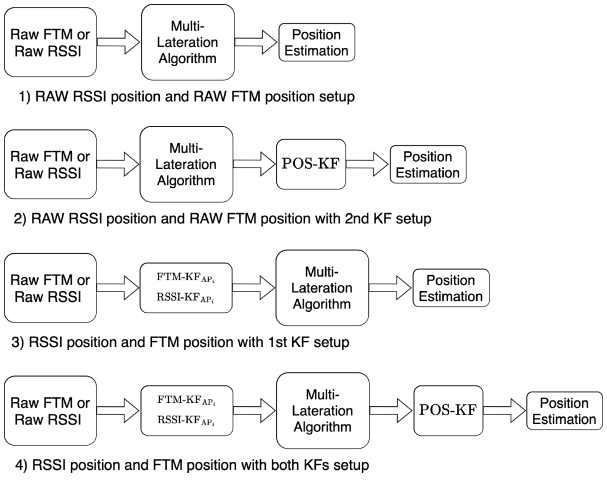


Fig. 9. Four different system setups adopted for both RSSI- and FTM-based localization experimental evaluation.

1) without any KF applied to the input RSSI and FTM data (no filtering): the corresponding setup is denoted as *RAW RSSI position* and *RAW FTM position*.
2) with only the POS-KF applied to the raw RSSI and raw FTM data (single filtering): the corresponding setup is denoted as *RAW RSSI position with second KF* and *RAW FTM position with second KF*.
3) with only the RSSI-KF$_{AP_i}$ and FTM-KF$_{AP_i}$ filters, $i = 1, \ldots, 6$, applied to the raw input data before multilateration (single filtering): the corresponding setup is denoted as *RSSI position with first KF* and *FTM position with first KF*.
4) with both the double KFs applied at the raw input RSSI and FTM data, as well as after multilateration (double filtering): the corresponding setup is denoted as *RSSI position with both KFs* and *FTM position with both KFs*.

1) Static UAV: Considering the RSSI-based approach, the *RAW RSSI position* setup achieves acceptable results, as shown in Fig. 10(a) and further confirmed by the empirical cumulative distribution function (ECDF), as shown in Fig. 10(c). From the obtained results, one can notice the beneficial effects of the *RAW RSSI position with second KF* filtering setup, which reduces the overall positioning error, lowering the *raw* RSSI mean position error from 2.58 m (no filtering) to 2.23 m, as given in Table I. The error reduction is higher for the 95th percentile position error, which reduces from 4.61 m (no filtering) to 3.26 m.

With regard to the double filtering setup, in Fig. 10(b), it can be observed that the point cloud of the *RSSI position with both KFs* setup concentrate in an approximate circular area, near the true UAV GNSS coordinates, which has a radius shorter than that of the point cloud associated with the *RSSI position with first KF*. This is also confirmed by

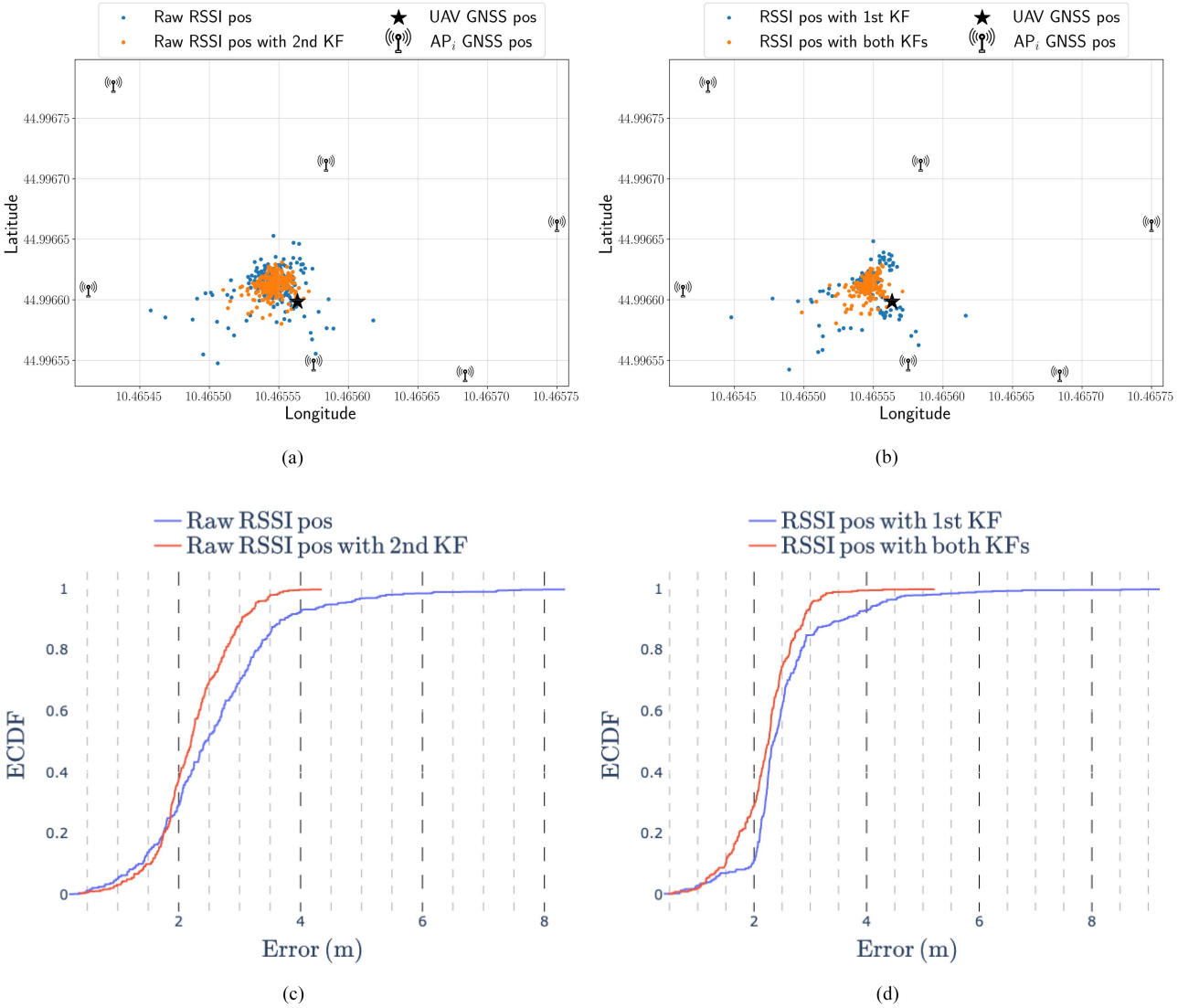


Fig. 10. Evaluation of the implemented Wi-Fi-based localization system in the static hovering scenario with different filtering setups. (a) Experimental position point clouds with *RAW RSSI position* and *RAW RSSI position with second KF* setups. (b) Experimental position point clouds with *RSSI position with first KF* and *RSSI position with both KFs* setups. (c) ECDF of the position error with *RAW RSSI position* and *RAW RSSI position with second KF* setups. (d) ECDF of the position error with *RSSI position with first KF* and *RSSI position with both KFs* setups.

TABLE I
Experimental Evaluation of RSSI- and FTM-Based Localization With the UAV in Static Hovering Conditions With Different Filtering Setups

Filtering Method	Mean Position Error [m]	95-th Percentile Error [m]
RAW RSSI	2.58	4.61
RAW RSSI with 2nd KF	2.23	3.26
RSSI with 1st KF	2.54	4.20
RSSI with both KFs	2.21	3.13
RAW FTM	1.99	5.82
RAW FTM with 2nd KF	1.51	3.57
FTM with 1st KF	1.70	4.81
FTM with both KFs	1.41	3.22

the ECDF, as shown in Fig. 10(d), as well as by the values listed in Table I: the mean positioning error reduces from 2.54 m (single filtering) to 2.21 m (double filtering), whereas the 95th percentile positioning error reduces from 4.20 m (single filtering) to 3.13 m (double filtering).

Regarding the FTM-based approach, the *RAW FTM position* setup achieves a more consistent and better result with respect to the RSSI-based double KF setup, as given in Table I, with a mean positioning error equal to 1.99 m. However, it is interesting to look at the higher (with respect to the RSSI-based system) 95th percentile error of the *RAW FTM position* setup, equal to 5.82 m, which is the overall worst. Comparing the *RAW FTM position* setup with the *RAW FTM position with second KF* setup, the latter allows to lower the mean positioning error to 1.51 m and the 95th percentile to 3.57 m, as confirmed by the ECDFs shown in Fig. 11(c) and visually illustrated by the point clouds in Fig. 11(a).

With regard to the *FTM position with both KFs* (double filtering) setup, Fig. 11(b) shows the filtered estimated

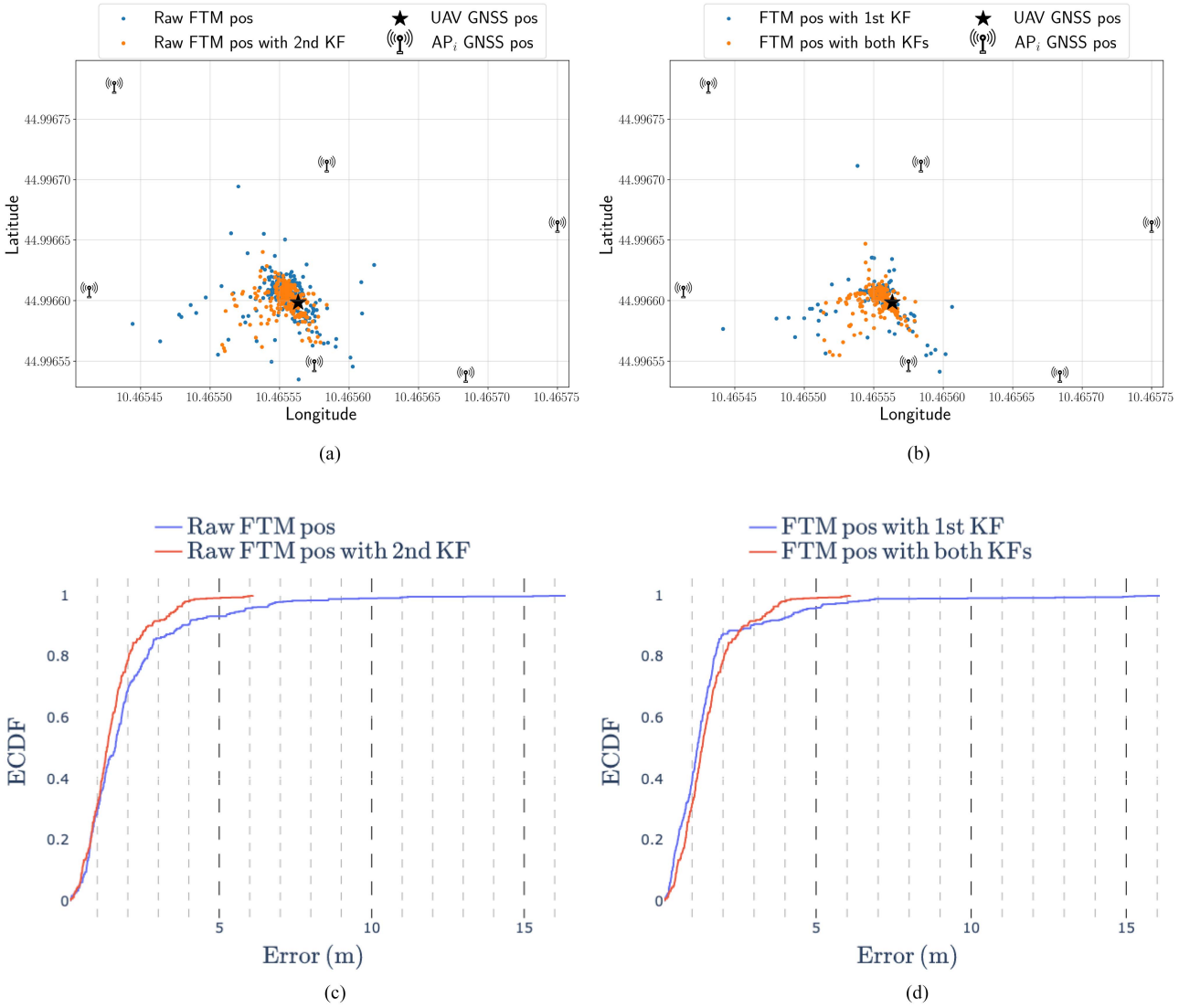


Fig. 11. Evaluation of the implemented Wi-Fi-based localization system in the static hovering scenario with different filtering setups. (a) Experimental position point clouds with *RAW FTM position* and *RAW FTM position with second KF* setups. (b) Experimental position point clouds with *FTM position with first KF* and *FTM position with both KFs* setups. (c) ECDF of the position error with *RAW FTM position* and *RAW FTM position with second KF* setups. (d) ECDF of the position error with *FTM position with first KF* and *FTM position with both KFs* setups.

positions concentrate nearer the true UAV GNSS position with respect to estimated positions in Fig. 11(a). This is confirmed by the ECDFs shown in Fig. 11(d) and by the values listed in Table I, where the *FTM position with both KFs* setup achieves a mean positioning error equal to 1.41 m, with a 95th percentile error equal to 3.22 m, thus achieving slightly better results than all the other aforementioned RSSI-based and FTM-based filtering setups. For the sake of comparison, the *RSSI position with first KF* only achieves a mean positioning error equal to 1.70 m, with a 95th positioning error of 4.81 m (the second worst value for this metric in the static scenario). The obtained FTM results are significantly better than the expected *rr* for the Wi-Fi 2.4 GHz 40 MHz technology discussed in Section III-B.

As expected, in the static scenario, the FTM-based localization solution manages to achieve the best performance with or without filters with respect to RSSI-based solutions, with the double KF setups able to further improve the localization accuracy of both the RSSI and FTM systems, but with the relatively higher gain in the FTM solution.

2) Mobile UAV: In the considered scenario with the quadcopter flying along a square path, both RSSI- and FTM-based solutions achieve (as expected) worse results with respect to those experienced in the static scenario. Starting with the RSSI-based approach, the *RAW RSSI position* setup significantly suffers from the movement of the UAV, as shown in Fig. 12(a) and further confirmed by the values listed in Table II. More in detail, the unfiltered localization solution achieves a mean positioning error equal to 5.49 m, with a 95th percentile error equal to 9.99 m. According to the ECDF, as shown in Fig. 12(c), it is possible to observe the beneficial effects of the *RAW RSSI position with second KF* setup, which marginally reduces

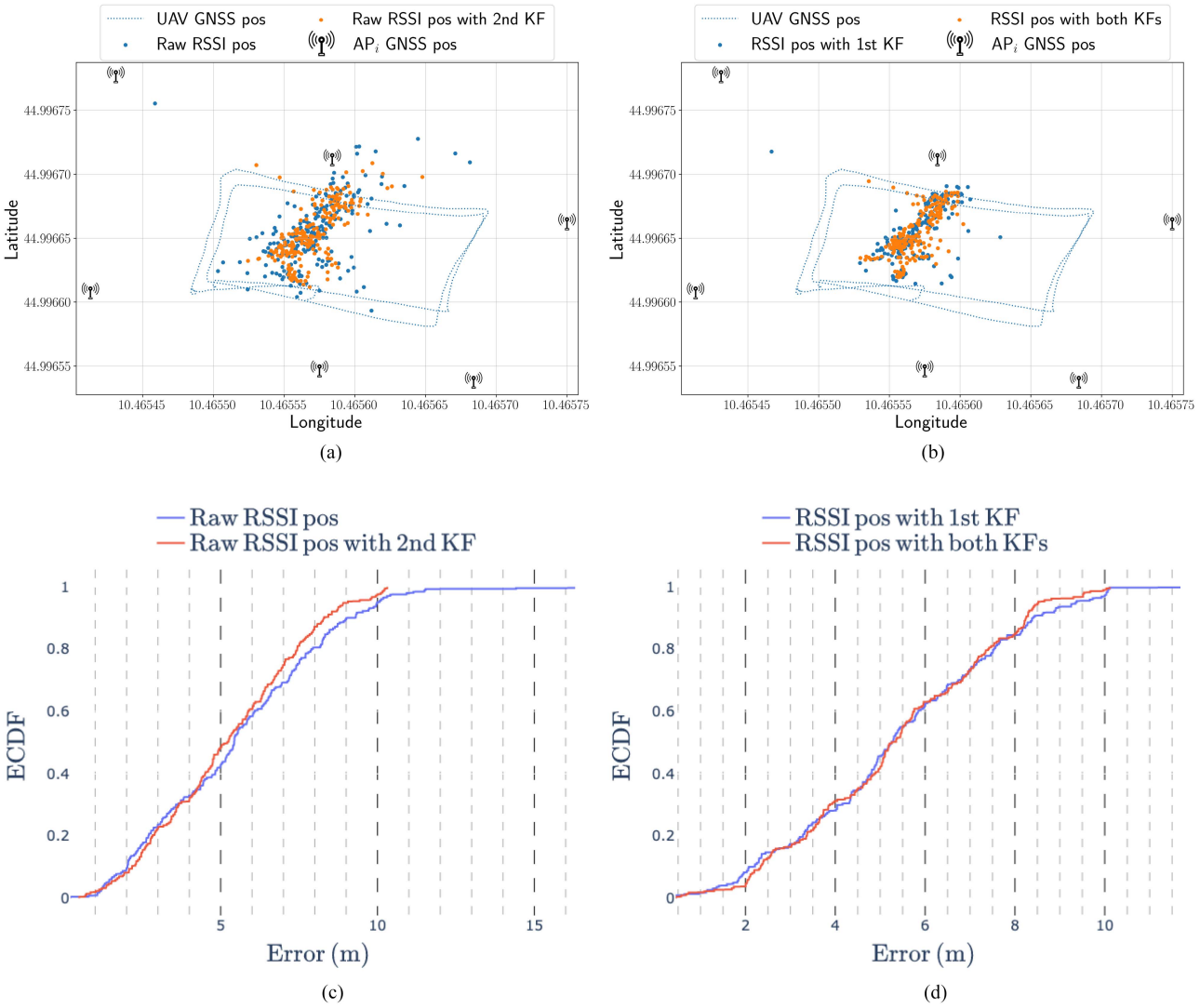


Fig. 12. Evaluation of the implemented Wi-Fi-based localization system in the mobile scenario with different filtering setups. (a) Experimental position point clouds with *RAW RSSI position* and *RAW RSSI position with second KF* setups. (b) Experimental position point clouds with *RSSI position with first KF* and *RSSI position with both KFs* setups. (c) ECDF of the position error with *RAW RSSI position* and *RAW RSSI position with second KF* setups. (d) ECDF of the position error with *RSSI position with first KF* and *RSSI position with both KFs* setups.

TABLE II
Experimental Evaluation of RSSI- and FTM-Based Localization With the UAV in Mobile Conditions With Different Filtering Setups

Filtering Method	Mean Position Error [m]	95-th Percentile Error [m]
RAW RSSI	5.49	9.99
RAW RSSI with 2nd KF	5.23	9.06
RSSI with 1st KF	5.37	9.35
RSSI with both KFs	5.34	8.52
RAW FTM	3.76	6.62
RAW FTM with 2nd KF	3.32	5.63
FTM with 1st KF	3.34	5.73
FTM with both KFs	3.30	5.70

the mean error and 95th percentile error to 5.23 and 9.06 m, respectively.

Considering the *RSSI position with first KF* and *RSSI position with both KFs* setups, none of these approaches manages to improve the results of the *RAW RSSI with second KF* setup, as confirmed by the mean error, given in Table II and close to 5.35 m in both cases, and by the ECDFs shown in Fig. 12(b). In any case, regardless of the filtering approach applied to the RSSI-based localization system, it is clearly impossible to understand (relying only on the RSSI-based estimated coordinates) in which part of the predefined flight path the UAV is located. Therefore, this makes the use of this solution for localization purposes with a mobile target unfeasible.

Significantly better results are achieved by the FTM-based systems. As for the static hovering scenario, the *RAW FTM position* setup leads to better results with respect to all the RSSI-based solutions, as listed in Table II. More in detail, without filtering, the FTM-based system achieves a mean positioning error equal to 3.76 m, with a 95th percentile error equal to 6.62 m. The unfiltered FTM-estimated

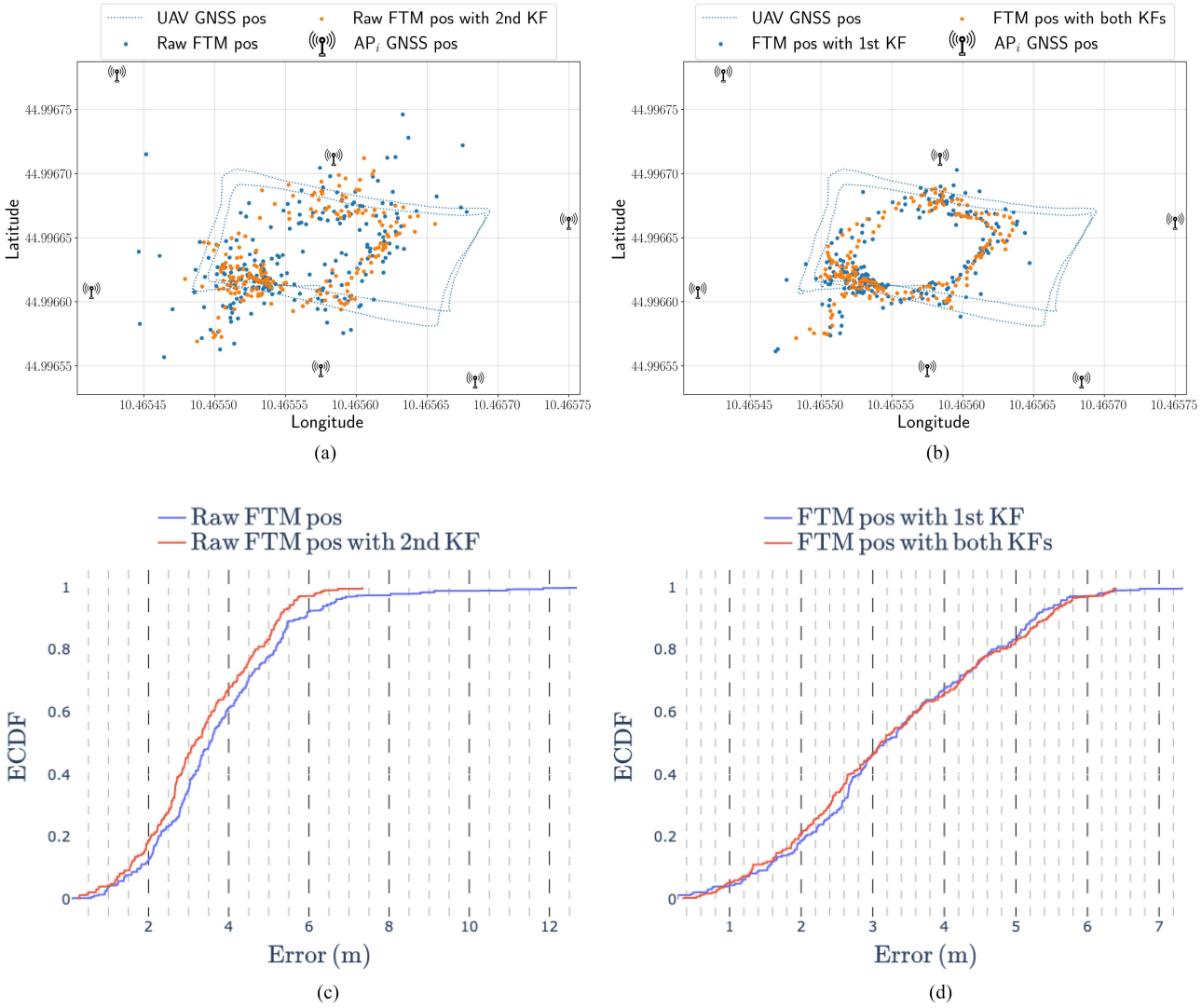


Fig. 13. Evaluation of the implemented Wi-Fi-based localization system in the mobile scenario with different filtering setups. (a) Experimental position point clouds with *RAW FTM position* and *RAW FTM position with second KF* setups. (b) Experimental position point clouds with *FTM position with first KF* and *FTM position with both KFs* setups. (c) ECDF of the position error with *RAW FTM position* and *RAW FTM position with second KF* setups. (d) ECDF of the position error with *FTM position with first KF* and *FTM position with both KFs* setups.

positions are shown (as a point cloud) in Fig. 13(a), together with the estimated positions computed adopting the *RAW FTM position with second KF* setup. As highlighted in the ECDFs shown in Fig. 13(c), the adoption of a KF after the application of a multilateration algorithm, as in the *RAW FTM position with second KF* setup, allows to reduce the mean positioning error and the 95th percentile error to 3.32 and 5.63 m, respectively.

Considering the *FTM position with both KFs* (double filtering) setup, Fig. 13(b) shows a smaller number of outlier positions as well as the overall point cloud closer to the ground truth GNSS-based flight path followed by the UAV. This seems valid for both the *FTM position with first KF* and *FTM position with both KFs* setups, as confirmed by the similar ECDFs shown in Fig. 13(d) and the values listed in Table II. In fact, both prefiltering and double filtering approaches allow to achieve similar mean positioning errors equal to 3.34 and 3.30 m, respectively, providing a limited improvement with respect to the *RAW FTM position with second KF* setup. Then, similar 95th percentile positioning errors are confirmed by the values listed in Table II. In this case, the FTM-based localization results are aligned with the expected *rr* for the Wi-Fi 2.4 GHz 40 MHz technology discussed in Section III-B.

With regard to the mobile scenario, despite the discussed positioning errors, it must be taken into account that the goal of the proposed Wi-Fi RSSI- and FTM-based localization system is to provide an approximate knowledge of the position of the UAV, possibly flying in a BVLOS condition without a (or with a weak) GNSS signal reception. Therefore, behind the average positioning error (in terms

of values), it is also relevant to evaluate the trajectories of the ground truth GNSS-based path with respect to the Wi-Fi estimated positions of the proposed systems, in order to understand which approach can allow to better understand where the UAV is flying with respect to the its true position.

As shown in Figs. 12(a) or 12(b) and 13(a) or 13(b), related to the RSSI- and FTM-based positioning in the mobile scenario, respectively, the first two "images" do not provide any reliable information to understand where the UAV is positioned along the true trajectory, while in the latter two images, the estimated positions are approximately distributed along the true flight path. Therefore, the RSSI -based solutions are not suitable for real-time applicability to a UAV, at least with the proposed architecture and approach, while the FTM-based approach allows to estimate the UAV position in real time with significantly higher accuracy, even in mobile conditions. The best performance is achieved by the double filtering setup: the estimated positions, as shown in Fig. 13(b), are mostly distributed along the true flight path, indicating that the corresponding system can provide an approximate position estimate of the UAV in the case of GNSS outage, thus allowing the pilot to understand where the UAV is flying along the planned path.

C. Relative Positioning Error Evaluation

Another relevant metric to be used for evaluating the performance of the proposed Wi-Fi-based localization system is the relative positioning error with respect to the average target–AP distance, as proposed in [9]. This allows to characterize the performance as a function of the "shape" of the considered scenario, regardless of its actual scale.

According to this evaluation metric, in the *static scenario*, the average UAV–AP distance, with the UAV hovering in a fixed position, is approximately equal to $\overline{d}_{\text{AP}}^{\text{STATIC}} = 13.42$ m. Considering the RSSI-based localization system with the double KFs setup (which, according to Table I, achieves the lowest mean positioning error equal to 2.21 m), the relative position error (with respect to $\overline{d}_{\text{AP}}^{\text{STATIC}}$) is approximately equal to 16.67%. Considering instead the FTM-based system with the double KFs setup (which, according to Table I, achieves the lowest mean positioning error equal to 1.41 m), the relative position error is approximately equal to 10.65%.

Focusing on the *mobile scenario*, the mean AP–target distance is approximately equal to $\overline{d}_{\text{AP}}^{\text{MOBILE}} = 14.01$ m. Therefore, according to Table II, for the *RAW RSSI with second KF* setup, the relative position error (with regard to $\overline{d}_{\text{AP}}^{\text{MOBILE}}$) is approximately equal to 37.36%, while for the FTM-based solution with both KFs, the relative position error is approximately equal to 23.56%.

In both static and mobile scenarios, it can be concluded that the FTM-based approach with both KFs reduces the relative position error observed with RSSI-based approach by approximately 37%.[4]

D. Update Frequency and Processing Times

Finally, since the system has been designed for real-time localization purposes, the measured position update rate must be evaluated to further understand the capabilities of the proposed solutions. As mentioned in Section V, with six APs deployed in the experimental environment acting as FTM *responders*, the ESP32S3 can provide a list of Wi-Fi scan results with six FTM measurements and all the RSSIs of the nearby APs approximately every $\theta_{\text{cycle}} = 522$ ms. More in detail, θ_{cycle} is defined as

$$\theta_{\text{cycle}} = \upsilon_{\text{wifi}} + n \cdot \nu \tag{41}$$

where υ_{wifi} is the Wi-Fi scan time [dimension: (ms)], approximately equal to 330 ms; $n = 6$ is the number of APs; and ν is the FTM *Burst* duration [dimension: (ms)], which, given the parameters adopted in the proposed architecture, is equal to 32 ms.

Therefore, considering the measured processing time introduced by the RPi4-handled filtering and multilateration, denoted as $\varkappa$ [dimension: (ms)] and approximately equal to 66 ms, the overall update period increases to approximately $\Theta_{\text{cycle}} = 588$ ms. This leads to a position update rate equal to $f_{\text{pos}} = 1/\Theta_{\text{cycle}} \approx 1.7$ Hz, which is sufficient for several real-time UAV applications.

E. Complexity

The complexity of the proposed system mainly corresponds to the computational complexity required to apply the KFs at each Wi-Fi AP, as well as the computation of the multilateration algorithm solved through the LSMR algorithm. Thus, it might be argued that the system involves several steps, each of them with its own computational complexity, which can be quantified as follow.

Collecting RSSI and FTM measurements from a number, denoted as A, of Wi-Fi APs is a linear scan operation through the APs' list, thus resulting in a complexity of $O(A)$. Subsequently, as detailed in Section V, for each AP's data, one KF is executed on the RSSI values (RSSI-KF$_{\text{AP}_i}$) and one on the FTM values (FTM-KF$_{\text{AP}_i}$), to remove possible noise in the measurements. According to the literature [48], a linear KF has a complexity of $O(n^3)$, where n is the size of the state vector in our filters. This is because the most computationally intensive step in a KF involves matrix multiplications and inversions, which generally have a complexity of $O(n^3)$. Therefore, since each AP requires two KFs, the resulting complexity is equal to $O(A \cdot 2n^3) = O(2An^3) = O(An^3)$.

Then, the next step—the execution of the multilateration algorithm, solved with the LSMR approach (as discussed in Section III-C and in [48])—involves matrix multiplication

[4]This value is obtained as $[(16.67 - 10.65)/16.67] \cdot 100$ in the static scenario and as $[(37.36 - 23.56)/37.36] \cdot 100$ in the mobile scenario.

with its transpose, thus corresponding to a complexity of $O(A^3)$.

After obtaining the position estimates from the multilateration, the POS-KFs are applied to both RSSI and FTM-based positions. Hence, since each KF has a complexity of $O(n^3)$ (as detailed before), this task results in a complexity of $O(2n^3) = O(n^3)$.

In the end, it is possible to estimate the overall complexity of the proposed algorithm by simply adding the complexity terms of each step, i.e.: $O(A) + O(An^3) + O(A^3) + O(n^3)$. The dominant term in this sum, especially for typical scenarios where $A \geq 3$, is $O(A^3)$, which is also the overall complexity of the described Wi-Fi-based localization system.

F. System Limitations

Although the performance of the proposed low-cost system architecture might be suitable for some UAV applications where the achieved accuracy level is sufficient to compensate for temporary GNSS outages, the proposed solution cannot provide submeter positioning capabilities, as well as more complex pose estimation of the UAV, as achieved in [27] and [28]. As shown by our solution, the results in [27] and [28] also confirm the limitations of Wi-Fi-based positioning, which requires LOS (between Wi-Fi APs and the UAV) to avoid inaccurate distance estimation and, thus, large position estimations errors.

Another limitation of the proposed solution is the operational area, which, even if wider than other positioning solutions (such as UWB based), since it does also not need to synchronize the clocks of the deployed Wi-Fi anchors, keeping the infrastructure cost lower, cannot compete with other on-board positioning solutions, such as visual simultaneous localization and mapping, more suitable to cover large environments without any infrastructure, as proposed in [49]. To this end, a deeper discussion about the scalability of our proposed solution is detailed in Section VI-G.

Finally, due to the low-cost nature of the proposed system, another limitation could be related to the usage of limited bandwidth in the 2.4-GHz spectrum band, which limits the accuracy capabilities of the system itself. As detailed in the future research directions discussed in Section VII, the usage of additional spectrum in the 5-GHz Wi-Fi band might allow to achieve more consistent submeter level accuracy, despite a slightly higher cost and reduced operational range.

G. Scalability

Finally, as mentioned in Section VI-F, the operational area of the proposed localization system might be a limiting factor. However, given the absence of need to synchronize the clocks of the deployed Wi-Fi anchors, it can be stated that our proposed approach is scalable since, as shown in Section VI-C, our solution provides a localization accuracy which is, on average, a fixed percentage of the distance between the UAV and a Wi-Fi AP. Therefore, it is possible to naturally scale the system by increasing the distance between UAV and APs. Nevertheless, increasing too much the distance is not compatible with typical transmission power levels of Wi-Fi transceivers. Should the transmit power be increased, our solutions would directly scale.

Another effective approach to make our solution scalable would be to increase the number of APs deployed in the area of interest, so that, on the average, the UAV has enough APs within the transmission range, thus guaranteeing the obtained performance.

VII. IMPROVEMENTS

Several approaches might be implemented to further enhance the performance of the proposed Wi-Fi-based localization system. In the following, we outline a few relevant ones identified for future research activities.

1) Exploit the 5-GHz Wi-Fi band for RSSI-based distance estimation purposes: the higher carrier frequency provides more granular measurements.
2) Exploit other Wi-Fi-related signal quality metrics, such as the CSI, which can be used to estimate the AoA on Wi-Fi multiple-input/multiple-output systems, thus increasing the position estimation accuracy [27], [28]. This, however, comes at the cost of more expensive devices and a higher complexity of the system architecture.
3) Utilization of 5 GHz, 80 MHz, or 160 MHz Wi-Fi FTM equipments, in order to reduce the rr and achieve a submeter level localization positioning error.
4) Adoption of an EKF instead of a KF, as well as other alternative filters, with the purpose of enhancing the filtering capabilities of the proposed solution, thus improving the final localization accuracy.
5) Adoption of a different algorithm for the localization task instead of the multilateration solution, such as, for example, the two-step M-estimator [50], which might be more effective in removing the noise, thus allowing to achieve a smaller positioning error with respect to the multilateration solution adopted in the solution proposed here.
6) Evaluation of sensor fusion-oriented approaches to exploit both RSSI- and FTM-based ranging solutions, aiming at developing LOS and NLOS channel estimation models suitable to improve the outliers removal filter and, therefore, to enhance the distance estimation, eventually increasing the overall positioning accuracy.

VIII. CONCLUSION

The proposed real-time Wi-Fi-based localization system (discussed in Section III) has been implemented and evaluated on a real UAV, aiming at understanding

the localization performance achievable with low-cost IEEE 802.11mc 2.4-GHz COTS hardware, as well as adopting traditional and well-known filtering and ranging estimation techniques (discussed in Sections IV and V). The obtained accuracy with the FTM-based solution in mobile conditions (presented in Section VI) is already sufficient in environments with weak GNSS or GNSS-denied conditions, where the adoption of UWB might be too expensive or infeasible (given the larger area) and not necessary (since sub-30 cm might be needed only on specific missions involving small UAVs, but not in BVLOS missions).

The proposed solution is effective as temporary GNSS backup solution to provide the pilot with an approximate position of the UAV. Moreover, the complexity and scalability analyses and the system limitations, as well as the identified improvements (detailed in Section VII) might further enhance the performance of the proposed localization system, thus paving the way to new applications. In all cases, we expect that the complexity and, therefore, the cost of the infrastructure will remain low, since FTM is being supported by most of the newly released devices, even on the 5-GHz band (which should provide significantly better localization capabilities).

APPENDIX A
WI-FI SCANNING DATA RETRIEVAL

As mentioned in Section V, in the following, additional details on the Wi-Fi scanning data retrieval are presented. Once the Wi-Fi scan results are obtained by the ESP32S3 module connected to the RPi4 on-board the quadcopter, they are then parsed into a JSON string and sent, through the USB-enabled serial port, to the RPi4 itself, with the structure of the generated JSON string depending on the result of the Wi-Fi scan process.

If the scanned AP is an FTM *responder*, then the JSON string includes the FTM-related fields, as follows:

```
{"SSID": "FTM_1", "MAC":
"f6:12:fa:5a:05:10",
"rtt_est": 346, "rtt_raw": 478,
"dist_est": 5200,
"num_frames": 15, "mean_rssi": -
45.00}
```

where `SSID` corresponds to the name of the Wi-Fi network; the `MAC` address is the unique physical address of the AP creating its own Wi-Fi network; `rtt_est` is the $\overline{\text{RTT}}$ between the *responder* and the *initiator* [dimension: (ns)] taking into account the clock drift compensation; `rtt_raw` is the *raw* $\overline{\text{RTT}}$ between the *responder* and the *initiator* [dimension: (ns)]; `dist_est` is the average estimated distance $\overline{d}$ between the *responder* and the *initiator* [dimension: (cm)], derived from `rtt_est` and without the initial offset correction; `num_frames` is the total number of exchanged frames (i.e., including both *FTM* and *ACK* frames) during the FTM process; and `mean_rssi` is the average RSSI [dimension: (dBm)] measured over all the exchanged FTM frames.

Otherwise, in the case that the scanned AP is *not* an FTM *responder*, then only `SSID`, `MAC` address, and `RSSI` are included as follows.

```
{"SSID": "Tenda", "MAC":
"f2:5a:01:5a:12:f4",
"mean_rssi": -52.00}
```


ACKNOWLEDGMENT

The ECSEL/KDT JUs received support from the European Union's Horizon 2020/Horizon Europe research and innovation programme and the nations involved in the mentioned projects. The work reflects only the authors' views; the European Commission is not responsible for any use that may be made of the information it contains.

REFERENCES

[1] Y. Zhao, X. Li, Y. Ji, and C.-Z. Xu, "Wireless power-driven positioning system: Fundamental analysis and resource allocation," *IEEE Internet Things J.*, vol. 6, no. 6, pp. 10421–10430, Dec. 2019, doi: 10.1109/JIOT.2019.2939215.

[2] J. P. Queralta, C. M. Almansa, F. Schiano, D. Floreano, and T. Westerlund, "UWB-based system for UAV localization in GNSS-Denied environments: Characterization and dataset," in *Proc. 2020 IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Las Vegas, NV, USA, 2020, pp. 4521–4528, doi: 10.1109/IROS45743.2020.9341042.

[3] A. Alarifi et al., "Ultra wideband indoor positioning technologies: Analysis and recent advances," *Sensors*, vol. 16, pp. 1–36, 2016, doi: 10.3390/s16050707.

[4] U. Seidaliyeva et al., "Advances and challenges in drone detection and classification techniques: A state-of-the-art review," *Sensors*, vol. 24, no. 1, 2024, Art. no. 125, doi: 10.3390/s24010125.

[5] M. A. Khan, H. Menouar, A. Eldeeb, A. Abu-Dayya, and F. D. Salim, "On the detection of unauthorized drones—Techniques and future perspectives: A Review," *IEEE Sensors J.*, vol. 22, no. 12, pp. 11439–11455, Jun. 2022, doi: 10.1109/JSEN.2022.3171293.

[6] H. Friis, "A note on a simple transmission formula," *Proc. IRE*, vol. 34, no. 5, pp. 254–256, May 1946, doi: 10.1109/JRPROC.1946.234568.

[7] S. Sadowski and P. Spachos, "RSSI-Based indoor localization with the Internet of Things," *IEEE Access*, vol. 6, pp. 30149–30161, 2018, doi: 10.1109/ACCESS.2018.2843325.

[8] C. Yang and H.-R. Shao, "WiFi-based indoor positioning," *IEEE Commun. Mag.*, vol. 53, no. 3, pp. 150–157, Mar. 2015, doi: 10.1109/MCOM.2015.7060497.

[9] F. Carpi et al., "Experimental analysis of RSSI-based localization algorithms with NLOS pre-mitigation for IoT applications," *Comput. Netw.*, vol. 225, Apr. 2023, Art. no. 109663, doi: 10.1016/j.comnet.2023.109663.

[10] R. M. M. R. Rathnayake et al., "RSSI and machine learning-based indoor localization systems for smart cities," *Eng*, vol. 4, no. 2, pp. 1468–1494, May 2023, doi: 10.3390/eng4020085.

[11] N. Singh, S. Choe, and R. Punmiya, "Machine learning based indoor localization using Wi-Fi RSSI fingerprints: An overview," *IEEE Access*, vol. 9, pp. 127150–127174, 2021, doi: 10.1109/ACCESS.2021.3111083.

[12] A. Poulose and D. S. Han, "Performance analysis of fingerprint matching algorithms for indoor localization," in *Proc. 2020 Int. Conf. Artif. Intell. Inf. Commun. (ICAIIC)*, Fukuoka, Japan, 2020, pp. 661–665, doi: 10.1109/ICAIIC48513.2020.9065220.

[13] B. R. Stojkoska, J. Palikrushev, K. Trivodaliev, and S. Kalajdziski, "Indoor localization of unmanned aerial vehicles based on RSSI," in *Proc. IEEE 17th Int. Conf. Smart Technol. (EUROCON)*, Ohrid, Macedonia, Aug. 2017, pp. 120–125, doi: 10.1109/EUROCON.2017.8011089.

[14] A. Booranawong, P. Thammachote, and Y. Sasiwat, "Real-time tracking of a moving target in an indoor corridor of the hospital building using RSSI signals received from two reference nodes," *Med. Biol. Eng. Comput.*, vol. 60, pp. 439–458, Jan. 2022, doi: 10.1007/s11517-021-02489-6.

[15] A. Poulose, J. Kim, and D. S. Han, "A sensor fusion framework for indoor localization using smartphone sensors and Wi-Fi RSSI measurements," *Appl. Sci.*, vol. 9, no. 20, 2019, doi: 10.3390/app9204379.

[16] H. Jin and P. Papadimitratos, "Off-the-shelf Wi-Fi indoor smartphone localization," in *Proc. 2022 17th Wireless On-Demand Netw. Syst. Serv. Conf. (WONS)*, Oppdal, Norway, Apr. 2022, pp. 1–4, doi: 10.23919/WONS54113.2022.9764448.

[17] "Wi-Fi RTT (IEEE 802.11mc)," Accessed: Jul. 10, 2024. [Online]. Available: https://source.android.com/devices/tech/connect/wifi-rtt

[18] R. Peng and M. L. Sichitiu, "Angle of arrival localization for wireless sensor networks," in *Proc. 2006 3rd Annu. IEEE Commun. Soc. Sensor Ad Hoc Commun. Netw.*, Reston, VA, USA, vol. 1, Jan. 2007, pp. 374–382, doi: 10.1109/SAHCN.2006.288442.

[19] P. E. Numan, H. Park, C. Laoudias, S. Horsmanheimo, and S. Kim, "Smartphone-based indoor localization via network learning with fusion of FTM/RSSI measurements," *IEEE Netw. Lett.*, vol. 5, no. 1, pp. 21–25, Mar. 2023, doi: 10.1109/LNET.2022.3226462.

[20] 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 INFOCOM 2020 - IEEE Conf. Comput. Commun.*, Toronto, ON, Canada, Aug. 2020, pp. 1857–1866, doi: 10.1109/INFOCOM41043.2020.9155464.

[21] Y. Yu, R. Chen, L. Chen, W. Li, Y. Wu, and H. Zhou, "A robust seamless localization framework based on Wi-Fi FTM / GNSS and built-in sensors," *IEEE Commun. Lett.*, vol. 25, no. 7, pp. 2226–2230, Jul. 2021, doi: 10.1109/LCOMM.2021.3071412.

[22] Y. Sugiyama, K. Kobayashi, and W. Chujo, "A study on indoor drone positioning using Wi-Fi RTT ranging," in *Proc. 2023 VTS Asia Pacific Wireless Commun. Symp. (APWCS)*, Tainan city, Taiwan, Sep. 2023, pp. 1–5, doi: 10.1109/APWCS60142.2023.10234042.

[23] V. B. Vales, O. C. Fernández, T. Domínguez-Bolaño, C. J. Escudero, and J. A. García-Naya, "Fine time measurement for the Internet of Things: A practical approach using ESP32," *IEEE Internet Things J.*, vol. 9, no. 19, pp. 18305–18318, Oct. 2022, doi: 10.1109/JIOT.2022.3158701.

[24] K. Ashraf and A. Ashok, "P2P-DroneLoc: Peer-to-peer localization for GPS-denied drones using camera and WiFi fine time measurement," in *Proc. 2022 IEEE Int. Conf. Adv. Netw. Telecommun. Syst. (ANTS)*, Gandhinagar, India, Aug. 2023, pp. 1–6, doi: 10.1109/ANTS56424.2022.10227756.

[25] Z. Shaikhanov, A. Boubrima, and E. W. Knightly, "FALCON: A. networked drone system for sensing, localizing, and approaching RF targets," *IEEE Internet Things J.*, vol. 9, no. 12, pp. 9843–9857, Jun. 2022, doi: 10.1109/JIOT.2022.3152380.

[26] M. Bullmann et al., "Comparison of 2.4 GHz WiFi FTM- and RSSI-Based indoor positioning methods in realistic scenarios," *Sensors*, vol. 20, no. 16, 2020, Art. no. 4515, doi: 10.3390/s20164515.

[27] G. Chi et al., "Wi-Drone: Wi-Fi-based 6-DoF tracking for indoor drone flight control," in *Proc. 20th Annu. Int. Conf. Mobile Syst., Appl. Serv., Ser. MobiSys'22*, Portland, Oregon, 2022, pp. 56–68, doi: 10.1145/3498361.3538936.

[28] M. Sulaiman, A. A. S. AlQahtani, H. Liu, and M. Binalhaj, "Utilizing Wi-Fi access points for unmanned aerial vehicle localization for building indoor inspection," in *Proc. 2021 IEEE 12th Annu. Ubiquitous Comput., Electron. Mobile Commun. Conf. (UEMCON)*, New York, NY, USA, 2021, pp. 0775–0779, doi: 10.1109/UEMCON53757.2021.9666544.

[29] O. Hashem, K. A. Harras, and M. Youssef, "Accurate indoor positioning using IEEE 802.11mc round trip time," *Pervasive Mobile Comput.*, vol. 75, Aug. 2021, Art. no. 101416, doi: 10.1016/j.pmcj.2021.101416.

[30] Y. Gu et al., "WiFi-fingerprint based indoor localization map," US Patent US9706366B2, Jul. 2017.

[31] T. S. Rappaport, *Wireless Communications: Principles and Practice*. Englewood Cliffs, NJ, USA: Prentice-Hall, 2002, ISBN: 978-0-470-86697-9.

[32] C. Balanis, *Antenna Theory: Analysis and Design* (Wiley-Interscience Series). New York, NY, USA: Wiley Interscience, 2005. [Online]. Available: https://books.google.se/books?id=2SJ0CgAAQBAJ

[33] *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), pp. 1–3534, 2016, doi: 10.1109/IEEESTD.2016.7786995.

[34] J. Wagner et al., "FMCW system aspects for multipath environments," in *Proc. 2011 8th Workshop Positioning, Navigation Commun.*, Dresden, Germany, Jul. 2011, pp. 89–93, doi: 10.1109/WPNC.2011.5961021.

[35] "ESP32-S3," Accessed: Jul. 10, 2024. [Online]. Available: https://www.espressif.com/en/products/socs/esp32-s3

[36] L. Jaulin, "5 - Instantaneous localization," in *Mobile Robotics*, L. Jaulin, Ed. Amsterdam, The Netherlands: Elsevier, 2015, pp. 171–196, doi: 10.1016/B978-1-78548-048-5.50005-X.

[37] T. Kariya and H. Kurata, *Generalized Least Squares*. Hoboken, NJ, USA: Wiley, 2004, ISBN: 978-0-470-86697-9.

[38] D. C. L. Fong and M. Saunders, "LSMR: An iterative algorithm for sparse least-squares problems," *SIAM J. Sci. Comput.*, vol. 33, no. 5, pp. 2950–2971, 2011, doi: 10.1137/10079687X.

[39] R. E. Kalman, "A new approach to linear filtering and prediction problems," *J. Basic Eng.*, vol. 82, no. 1, pp. 35–45, Mar. 1960, doi: 10.1115/1.3662552.

[40] G. Einicke and L. White, "Robust extended Kalman filtering," *IEEE Trans. Signal Process.*, vol. 47, no. 9, pp. 2596–2599, Sep. 1999, doi: 10.1109/78.782219.

[41] L. Kleeman, "Understanding and applying Kalman filtering," Dept. Elect. Comput. Syst. Eng., Monash Univ., Clayton, VIC, Australia, Jul. 2007. Accessed: Jul. 10, 2024. [Online]. Available: https://www.cs.cmu.edu/motionplanning/papers/sbp_papers/kalman/kleeman_understanding_kalman.pdf

[42] V. Kordic, *Kalman Filter*, Rijeka, Croatia: IntechOpen, 2010, doi: 10.5772/233.

[43] "T3-S3 V1.0 ESP32-S3 LoRa SX1280 2.4 G development board," Accessed: Jul. 10, 2024. [Online]. Available: https://www.lilygo.cc/products/t3s3-v1-0?variant=43043715678389

[44] Open Robotics, "Robot operating system (ROS)," Accessed: Jul. 10, 2024. [Online]. Available: http://www.ros.org/

[45] DMA WGS 84 Development Committee, "Department of defense World Geodetic System 1984," Defense Mapping Agency, Washington, DC, USA, Tech. Rep. ADA280358, 2nd Ed., 1991. Accessed: Jul. 10, 2024. [Online]. Available: https://apps.dtic.mil/sti/pdfs/ADA280358.pdf

[46] J. R. Clynch, "Earth coordinates," SAGE Publications, Tech. Rep. 1, Feb. 2006. Accessed: Jul. 10, 2024. [Online]. Available: https://bit.ly/45FnDmu

[47] F. Klinker, "Exponential moving average versus moving exponential average," *Mathematische Semesterberichte*, vol. 58, no. 1, pp. 97–107, Dec. 2010, doi: 10.1007/s00591-010-0080-8.

[48] B. P. Gibbs, *Advanced Kalman Filtering, Least-Squares and Modeling: A Practical Handbook*. Hoboken, NJ, USA: Wiley, 2011, doi: 10.1002/9780470890042.

[49] N. Stathoulopoulos et al., "Redundant and loosely coupled LiDAR-Wi-Fi integration for robust global localization in autonomous mobile robotics," in *Proc. 2023 21st Int. Conf. Adv. Robot. (ICAR)*, Abu Dhabi, UAE, 2023, pp. 121–127, doi: 10.1109/ICAR58858.2023.10406402.

[50] G. Zeng, B. Mu, J. Chen, Z. Shi, and J. Wu, "Global and asymptotically efficient localization from range measurements," *IEEE Trans. Signal Process.*, vol. 70, no. 1 pp. 5041–5057, Aug. 2022, doi: 10.1109/TSP.2022.3198167.

Emanuele Pagliari received the Dr. Ing. degree in communication engineering and the Ph.D. degree in information technologies from the Department of Information Engineering, University of Parma, Parma, Italy, in 2020 and 2024, respectively.

He is currently a Postdoctoral Researcher with the Robotics and AI Group, Department of Computer, Electrical and Space Engineering, Luleå University of Technology, Luleå, Sweden. His research interests include IoT, wireless communication systems, UAVs, and heterogeneous networking.

Luca Davoli (Member, IEEE) received the Dr. Ing. degree in computer engineering and the Ph.D. degree in information technologies with the Department of Information Engineering, University of Parma, Parma, Italy, in 2013 and 2017, respectively.

He is currently a nontenured Assistant Professor with the Internet of Things (IoT) Laboratory, Department of Engineering and Architecture, University of Parma. He is a Research Scientist with things2i Ltd., a spin-off of the University of Parma dedicated to IoT and smart systems. His research interests include IoT, pervasive computing, big stream, and software-defined networking.

Gianluigi Ferrari (Senior Member, IEEE) received the Laurea (summa cum laude) and Ph.D. degrees in electrical engineering from the University of Parma, Parma, Italy, in 1998 and 2002, respectively.

Since 2002, he has been with the University of Parma, where he is currently a Full Professor of telecommunications and also the Coordinator with the Internet of Things (IoT) Laboratory, Department of Engineering and Architecture. He is a Cofounder and President of things2i ltd., a spin-off of the University of Parma dedicated to IoT and smart systems. His current research interests include signal processing, advanced communication and networking, IoT, and smart systems.

Open Access provided by 'Università degli Studi di Parma' within the CRUI CARE Agreement