Robust Particle Filter for Accurate WiFi-Based Indoor Positioning in the Presence of Outlier-Corrupted Sensor Data

Mohamed Aizad Bin Mohamed Ghazali[1], Aroland Kiring[2*], Lyudmila Mihaylova[3],
Hoe Tung Yew[4], Seng Kheau Chung[5], Farrah Wong[6]
Faculty of Engineering, Universiti Malaysia Sabah, Kota Kinabalu, Sabah, Malaysia[1, 2*, 4, 5, 6]
Department of Automatic Control and System Engineering, University of Sheffield, Sheffield, UK[3]

***Abstract*—This study presents a comprehensive evaluation of an outlier-robust particle filter (RPF) designed to improve indoor positioning accuracy in complex environments with substantial measurement noise and outliers. The RPF's performance is benchmarked against a standard Particle Filter (PF) using both simulated and real-world datasets. Simulation results indicate that the RPF consistently outperforms the PF in indoor positioning particularly when sensor measurements contain outliers, achieving significant reductions in root mean square error (RMSE) for position, velocity, and acceleration estimation, with improvements of approximately 40.02%, 38.48%, and 65.80%, respectively. Real-world experiments, applying a calibrated log-normal path loss model to Wi-Fi received signal strength (RSS) data, further corroborate the RPF's effectiveness, demonstrating a 93.61% improvement in positioning accuracy compared to the PF. These findings highlight the RPF's robustness in delivering high accuracy, especially in environments with measurement outliers, establishing it as a reliable solution for indoor tracking in noisy sensor environments.**

***Keywords*—Complex environments; indoor positioning; measurement noise and outliers; RMSE reduction; robust particle filter**

I. Introduction

Indoor Positioning Systems (IPS) have become increasingly important in today's technology-driven world, enabling accurate positioning of people and objects within indoor spaces [1]. Unlike outdoor positioning, which has been revolutionized by the Global Positioning System (GPS) [2], indoor environments present unique challenges that GPS cannot effectively address. These challenges include signal loss due to walls and obstacles, multipath interference, and the complexity of vertical positioning in multi-story buildings. As a result, IPS has become a significant area of research, essential for applications such as location-based services (LBS), asset tracking, and indoor navigation [3]. Various technologies support IPS, each with its strengths and limitations. Common technologies include Bluetooth Low Energy (BLE), Infrared, Ultra-Wideband (UWB), Magnetic Fields, as well as Wi-Fi. BLE is known due to its low power consumption and suitability for proximity-based applications; however, although its precision is susceptible to signal interference and environmental conditions [4]. Infrared systems offer high accuracy but require a direct line of sight between the transmitter and receiver, making them less practical in dynamic environments [5]. UWB provides high precision but demands specialized hardware, making large-scale deployment costly [6]. Magnetic field-based positioning can function without additional infrastructure but is sensitive to environmental changes and may require frequent recalibration [7].

Among these technologies, Wi-Fi has emerged as a practical and widely used solution for IPS [8]. Operating at 2.4 GHz and 5 GHz frequency bands conforming to IEEE 802.11 protocol, Wi-Fi is common in residential, commercial, and industrial environments. This existing Wi-Fi infrastructure can be leveraged for indoor positioning, reducing the need for additional hardware. Wi-Fi's ability to penetrate obstacles enhances its versatility and cost-effectiveness for indoor applications [9]. To improve the accuracy of Wi-Fi-based IPS, techniques such as Received Signal Strength (RSS) fingerprinting [10], Time of Arrival (ToA) [11], and Angle of Arrival (AoA) [12] are employed. Although AoA and UWB methods achieve higher absolute accuracy, RSS-based IPS is prioritized due to its simplicity (no extra hardware), use of existing Wi-Fi infrastructure, and ease of deployment. However, challenges such as signal fluctuations due to multipath effects and interference from other electronic devices can impact reliability. Accurate position estimation in IPS relies heavily on processing sensor data, which is often affected by noise and outliers [13]. Filtering algorithms are used to refine raw sensor data and improve positioning accuracy. Common algorithms include Kalman Filters (KF), Extended Kalman Filters (EKF), Unscented Kalman Filters (UKF), and Particle Filters (PF).

The KF is optimal for linear systems with Gaussian noise, providing efficient and accurate state estimates [14]. However, real-world environments often involve non-linearities and non-Gaussian noise, limiting the KF's effectiveness. The EKF addresses non-linear systems by linearizing it around current estimates, but this approach can yield suboptimal results in highly non-linear settings [15]. The UKF improves on the EKF by employing deterministic sampling to better represent the state mean and covariance, enhancing performance in non-linear cases [16]. Both EKF and UKF assume Gaussian noise, which is not always realistic in indoor environments prone to outliers. PFs provide a flexible alternative for non-linear, non-Gaussian systems [17]. Using a set of weighted particles to represent the system state distribution, PFs handle diverse noise types and complex dynamics effectively. It operate through a recursive prediction-update process: particles are propagated based on system dynamics in the prediction step, and its weights are adjusted during the update step based on observed measurements. This adaptability makes PFs wellsuited for indoor positioning, where signal propagation is irregular due to obstacles, reflections, and multipath effects. Despite their robustness, PFs face challenges such as weight degeneracy, where most particles contribute minimally to the state estimate after several iterations [18]. Resampling helps mitigate this issue, but increases computational load [19]. Nonetheless, PFs' ability to model non-linearities and manage non-Gaussian noise conditions makes them an ideal choice for complex indoor environments, outperforming Kalman-based filters under such conditions [20].

PFs are widely used in robotics, target tracking, and autonomous vehicles due to their ability to represent state distributions accurately. However, this capability comes with challenges. Maintaining and updating a large number of particles is computationally intensive, making real-time applications or systems with limited processing power difficult to manage [21]. The process of propagating and re-weighting particles adds complexity, especially when rapid updates are required. Weight degeneracy, where most particles contribute minimally to the state estimate, further degrades performance, particularly in noisy environments or when measurements include outliers [22]. Resampling helps address weight degeneracy, but adds to the computational load. In IPS, PFs are effective in managing non-linearities and non-Gaussian noise but are sensitive to outliers in sensor measurements. Outliers, caused by signal reflections, interference, or hardware malfunctions, distort state estimation by skewing particle weights [23]. This can result in inaccurate state estimates, as particles with incorrect weights disproportionately influence the distribution. Outliers exacerbate weight degeneracy by concentrating the majority of the weight on a few particles, reducing the effective number of particles and degrading the filter's performance. It can also mislead the PF, causing particles to cluster around incorrect states and reducing diversity in the particle set. This loss of diversity limits the filter's ability to explore the true state space, compromising accuracy and reliability. Consequently, PFs are less effective in environments with frequent outliers, reducing their ability to provide consistent and precise positioning estimates.

Given these challenges, improving the robustness of PFs to effectively handle outliers is essential for enhancing indoor positioning accuracy. This requires developing strategies to detect and mitigate the impact of outliers, ensuring that the PF remains accurate even with contaminated data. To address these challenges, this study introduces the outlier-robust Particle Filter (RPF), which is designed to improve the accuracy and robustness of indoor positioning in the presence of outliers. The RPF utilizes an adaptive noise model that dynamically switches between Gaussian and Student's t-distribution noise models based on the inferred likelihood of outliers in the sensor measurements. Under normal conditions, where measurements mainly contain Gaussian noise, the filter employs a Gaussian distribution [24]. However, when outliers are detected, it switches to a Student's-T distribution [25], providing increased resilience against extreme values. Unlike prior robust PF approaches that rely solely on Student's-T noise models in [25] and [26], the RPF integrates the Singer acceleration motion model, which inherently handles non-linear movement by jointly modeling position, velocity, and acceleration, and a log-normal path-loss sensor model to convert RSS into distance. This two-fold design explicitly accounts for both Gaussian and heavy-tailed (non-Gaussian) measurement noise. The effectiveness of the proposed RPF is validated through simulations and real-world experiments conducted in residential indoor environments. These experiments demonstrate the RPF's superior performance in positioning accuracy, robustness against outliers, and its ability to handle non-linear movements and non-Gaussian noise, outperforming standard PF methods. By addressing key challenges such as measurement noise, weight degeneracy, and particle diversity in the presence of outliers, this study presents a more robust and reliable solution for Wi-Fi-based IPS. The RPF marks a significant advancement in indoor positioning, addressing the limitations of PFs in real-world environments.

The remainder of this study is organized as follows: Section II introduces the system model and notation, beginning with the Singer acceleration motion model defining the state vector, transition matrix, and control and noise inputs and then presenting the RSS-based path-loss measurement framework, which includes both the multivariate Gaussian distribution for nominal noise and the multivariate Student's-T distribution to handle outliers. Section III formulates the filtering problem by first reviewing the standard particle filter covering sampling, weight update, and normalization for nonlinear, non-Gaussian state estimation and then detailing the proposed outlier-robust particle filter, which employs a dual-noise likelihood framework to switch between Gaussian and Student's-T hypotheses and incorporates Metropolis resampling to preserve particle diversity, as summarized in Algorithm 2. Section IV presents experiments and results under Wi-Fi RSS measurements, describing simulated data scenarios including 2D trajectory generation, noise and outlier injection, and RMSE performance metrics and real-world experiments using the ESP-32-WROOM-32 in a residential indoor setting with calibrated log-normal path-loss and positioning accuracy comparisons. Section V summarizes the key findings highlighting the RPF's superior robustness and accuracy and outlines directions for future research.

A. Motivation and Contribution

This work addresses the critical need to enhance the robustness of PFs against outliers while maintaining computational efficiency, a key requirement for practical IPS. Traditional PFs struggle with non-Gaussian noise and measurement outliers, which can severely degrade performance. To overcome these challenges, the RPF is introduced, featuring an adaptive noise model that dynamically switches between Gaussian and Student's-T distributions based on outlier detection. This approach ensures resilience to extreme values while preserving accuracy. The contributions of this work are threefold. Firstly, the RPF is specifically designed to mitigate the adverse effects of outliers in Wi-Fi-based IPS, addressing a significant challenge in environments where non-Gaussian measurement noise is prevalent. Secondly, the performance of the RPF is rigorously evaluated through extensive testing on both simulated datasets and real-world experiments, demonstrating its superior accuracy and robustness compared to traditional methods. Thirdly, a calibrated log-normal path loss model is integrated into the algorithm to provide a more accurate representation of indoor signal propagation, further enhancing reliability and positioning accuracy.

II. System Model

Notations: Bold lowercase letters (e.g., $\mathbf{x}$) are used to represent vectors, while bold uppercase letters (e.g., $\mathbf{A}$) denote matrices. Scalars are indicated with standard italic letters (e.g., x). The symbol $\mathbb{R}$ represents the set of all real numbers, and $\mathbb{R}^{m\times n}$ signifies the space of $m \times n$ real matrices. The identity matrix is written as $\mathbf{I}$, and the determinant of a matrix $\mathbf{A}$ is expressed as $|\mathbf{A}|$. A Multivariate Gaussian distribution, fully defined by its mean μ and covariance matrix Σ, is represented as $\mathcal{N}(\mu, \Sigma)$. The transpose operation on a matrix or vector is indicated with the superscript T.

A. Motion Model

Consider the trajectory of a mobile user in a two-dimensional (2D) floor plan, described by the Singer state-space model [27], as in Eq. (1):

$$\mathbf{x}_k = \mathbf{A}\mathbf{x}_{k-1} + \mathbf{B}_u\mathbf{u}_k + \mathbf{B}_w\mathbf{w}_k, \tag{1}$$

where, $\mathbf{x_k} = [x_k, \dot{x}_k, \ddot{x}_k, y_k, \dot{y}_k, \ddot{y}_k]^T$ is a state vector that consists of the mobile user's coordinates (x_k, y_k), velocity $(\dot{x}_k, \dot{y}_k)$, and acceleration $(\ddot{x}_k, \ddot{y}_k)$, respectively at time step k. The parameter $\mathbf{A} \in \mathbb{R}^{6\times 6}$ is the state transition matrix, the vector u_k denotes the control input, which represents any externally applied input or maneuvering command to the system, $\mathbf{B}_u \in \mathbb{R}^{6\times 2}$ is the control input matrix, and $\mathbf{B}_w \in \mathbb{R}^{6\times 2}$ represents the noise matrix, given by Eq. (2) and Eq. (3),

$$\mathbf{A} = \begin{bmatrix} 1 & T & T^2/2 & 0 & 0 & 0 \\ 0 & 1 & T & 0 & 0 & 0 \\ 0 & 0 & \alpha & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & T & T^2/2 \\ 0 & 0 & 0 & 0 & 1 & T \\ 0 & 0 & 0 & 0 & 0 & \alpha \end{bmatrix}, \tag{2}$$

$$\mathbf{B}_u = \begin{bmatrix} T^2/2 & 0 \\ T & 0 \\ 0 & 0 \\ 0 & T^2/2 \\ 0 & T \\ 0 & 0 \end{bmatrix}, \quad \mathbf{B}_w = \begin{bmatrix} T^2/2 & 0 \\ T & 0 \\ 1 & 0 \\ 0 & T^2/2 \\ 0 & T \\ 0 & 1 \end{bmatrix}, \tag{3}$$

where, T is the discretization period and $\alpha = 1/\tau$ is the reciprocal of the maneuvering time constant τ. The parameter $\mathbf{w}_k \in \mathbb{R}^{2\times 1}$ accounts for the state noise, and follows a zero-mean multivariate Gaussian distribution, $\mathbf{w}_k \sim \mathcal{N}(0, \mathbf{Q_k})$, with state noise covariance matrix $\mathbf{Q_k} = \mathbb{E}(\mathbf{w}_k\mathbf{w}_k^T) = \sigma_w^2\mathbf{I}$. The motion model describes how the mobile user moves at time steps $k = \{1, 2, 3, ...\}$. It captures the dynamics of the mobile user by considering the stochastic nature of acceleration [28]. This is particularly useful in scenarios where the motion of the user cannot be precisely model and is subject to random variations. By modeling the acceleration as a random process, the motion model provides a more realistic representation of the user's movement in real-world scenarios.

B. Sensor Measurement Model

In indoor environments, GPS signals are unavailable because it cannot penetrate building structures effectively. To address this limitation, signals emitted by indoor infrastructure can be utilized to indirectly determine the position of a mobile user within an indoor environments. One practical approach involves using Wi-Fi access points (APs) as transmitters, taking advantage of the existing Wi-Fi infrastructure in indoor spaces, while mobile devices function as receivers that measure the RSS from these APs [29]. The RSS values indicate the signal power received by the mobile device from each AP, which can be used to estimate the distance between the APs and the mobile user [30]. By combining RSS measurements from multiple APs, the position of the mobile device can be triangulated. The relationship between the RSS measurements and the distance is typically modeled using the Path Loss Model, which provides a mathematical framework for representing Wi-Fi signal propagation, given by [31],

$$z_k = z_0 + 10\beta \log_{10}(d_k) + v_k, \tag{4}$$

where, z_k is the RSS measurement at time step k at the receiver node, z_0 is the reference path loss at a 1-meter distance, β is the path loss exponent, d_k is the distance between the receiver and transmitter at time step k, and v_k is the measurement noise. For practical applications, a minimum of three APs is required for positioning, as this allows for triangulation to determine the position of the mobile user in the network. Thus, Eq. (4) can be expressed in vector form as $\mathbf{z}_k = [z_k^1, z_k^2, \ldots, z_k^n]^T$, where n refers to the number of APs.

1) Multivariate Gaussian Distribution: The measurement noise in Eq. (4) is often modeled as a zero-mean Gaussian distribution $v_k \sim \mathcal{N}(\mathbf{0}, \sigma_k^2)$. For multiple APs, the measurement noise is expressed as a vector $\boldsymbol{v}_k = [v_k^1, v_k^2, v_k^3, \ldots, v_k^n]^T$, where n denotes the total number of APs. Consequently, v_k follows a multivariate Gaussian distribution, $\boldsymbol{v}_k \sim \mathcal{N}(\mathbf{0}, \mathbf{R_k})$, with measurement noise covariance matrix $\mathbf{R_k} = \mathbb{E}[\boldsymbol{v}_k\boldsymbol{v}_k^T] = \sigma_v^2\mathbf{I}$. The probability density function (PDF) for a multivariate Gaussian distribution, characterized by a mean vector $\hat{\mathbf{z}}_k$ and a covariance matrix $\mathbf{R_k}$, can be expressed as Eq. (5):

$$p(\mathbf{z}_k) = (2\pi)^{-\frac{n}{2}}|\mathbf{R_k}|^{-\frac{1}{2}} \exp\Big(-\frac{1}{2}(\mathbf{z}_k - \hat{\mathbf{z}}_k)^T\mathbf{R}_k^{-1}(\mathbf{z}_k - \hat{\mathbf{z}}_k)\Big), \tag{5}$$

where, n represents the number of AP, $\mathbf{z}_k$ is the observed RSS measurement, $\hat{\mathbf{z}}_k$ is the predicted RSS measurement, $\mathbf{R}_k$ is the measurement noise covariance matrix, and $|\mathbf{R_k}|$ is the determinant of $\mathbf{R}_k$. It provides the likelihood of the RSS measurements given the estimated state of the system. The likelihood function is utilized in the update step of the PF to adjust particle weights based on how closely the predicted measurements align with the observed data.

2) Multivariate Student's-T Distribution: In cases, where sensor measurements contain outliers, the measurement noise in Eq. (4) follows a multivariate Student's-T distributions, $\mathbf{z}_k \sim St(\mathbf{0}, \mathbf{R}, \nu)$, with the PDF given by [26] Eq. (6),

$$p(\mathbf{z}_k) = \frac{\Gamma\left(\frac{\nu+n}{2}\right)}{\Gamma\left(\frac{\nu}{2}\right)(\nu\pi)^{n/2}\det(\mathbf{R}_k)^{1/2}} \times (1 + \frac{1}{\nu}(\mathbf{z}_k - \hat{\mathbf{z}}_k)^T\mathbf{R}_k^{-1}(\mathbf{z}_k - \hat{\mathbf{z}}_k))^{-\frac{\nu+n}{2}}, \tag{6}$$

where, n represents the total number of AP, $\mathbf{z}_k$ is the observed RSS measurement, $\hat{\mathbf{z}}_k$ is the predicted RSS measurement, $\mathbf{R}_k$ is the measurement noise covariance matrix, and ν is the degrees of freedom. The symbol $\Gamma(\cdot)$ represents the gamma function. The multivariate Student's-T distribution shares some similarities with the multivariate Gaussian distribution, especially when $\nu \geq 30$. As ν increases, the multivariate Student's-T distribution approaches the multivariate Gaussian distribution. A key distinction of the Student's-T distribution is its capacity to model heavier tails compared to the Gaussian distribution. The parameter ν controls the degrees of freedom and influences the thickness of the distribution's tails. Lower values of ν result in heavier tails.

III. Problem Formulation

A. Review of the Particle Filter

A PF is a state estimation method designed for non-linear systems, utilizing a collection of samples, known as particles, to approximate complex probability distributions. Unlike approaches constrained by Gaussian assumptions, PF can model arbitrary distributions. Within the context of indoor positioning, PF estimates the state of a mobile user, denoted as $\mathbf{x}_k$ in Eq. (1), by leveraging the RSS measurements represented as $\mathbf{z}_k$ in Eq. (4). The core objective of the PF is to derive the PDF that describes the relationship between the system states and measurements. To approximate the joint PDF at the prior time step $p(\mathbf{x}_{k-1}|\mathbf{z}_{1:k-1})$, the PF employs a weighted set of particles, $\{\mathbf{x}_{k-1}^{(i)}, i = 1, \ldots, N_p\}$ [32], given by Eq. (7):

$$p(\mathbf{x}_{k-1}|\mathbf{z}_{1:k-1}) \approx \sum_{i=1}^{N_p} w_{k-1}^{(i)} \delta(\mathbf{x}_{k-1} - \mathbf{x}_{k-1}^{(i)}), \tag{7}$$

where, $\{w_{k-1}^{(i)}, i = 1, \ldots, N_p\}$ represents the weights associated with each particle, where N_p denotes the total number of particles, and $\delta(\cdot)$ signifies the Dirac delta function. The posterior PDF $p(\mathbf{x}_k|\mathbf{z}_{1:k})$ is defined as Eq. (8):

$$p(\mathbf{x}_k|\mathbf{z}_{1:k}) \propto p(\mathbf{z}_k|\mathbf{x}_k) \sum_{j=1}^{N_p} w_{k-1}^{(j)} p(\mathbf{x}_k|\mathbf{x}_{k-1}^{(j)}), \tag{8}$$

where, $p(\mathbf{z}_k|\mathbf{x}_k)$ is the measurement likelihood, and $p(\mathbf{x}_k|\mathbf{x}_{k-1}^{(j)})$ represents the prior transition model. If the particles are drawn from a proposal distribution $\mathbf{x}_k^{(i)} \sim q(\mathbf{x}_k|\mathbf{x}_{k-1}^{(i)}, \mathbf{z}_k)$, the weights are updated according to Eq. (9):

$$w_k^{(i)} \propto w_{k-1}^{(i)} \frac{p(\mathbf{z}_k|\mathbf{x}_k^{(i)}) p(\mathbf{x}_k^{(i)}|\mathbf{x}_{k-1}^{(i)})}{q(\mathbf{x}_k^{(i)}|\mathbf{x}_{k-1}^{(i)}, \mathbf{z}_k)}. \tag{9}$$

The proposal distribution is assumed to be equal to the state transition probability, i.e. [see Eq. (10)],

$$q(\mathbf{x}_k^{(i)}|\mathbf{x}_{k-1}^{(i)}, \mathbf{z}_k) = p(\mathbf{x}_k^{(i)}|\mathbf{x}_{k-1}^{(i)}), \tag{10}$$

based on the rationale that the state dynamics sufficiently capture the evolution of the system, and the measurement z_k does not provide additional information at the sampling stage. This assumption simplifies the weight update,

$$w_k^{(i)} \propto w_{k-1}^{(i)} p(\mathbf{z}_k|\mathbf{x}_k^{(i)}). \tag{11}$$

Algorithm 1 The Particle Filter (PF).

(1) Input: Initialization
  for $k = 0$ **do**
    for $i = 1, \ldots, N_p$ **do**
      Samples: $\{\mathbf{x}_0^{(i)} \sim q(\mathbf{x}_0)\}$.
      Set initial weights: $w_0^{(i)} = 1/N_p$.
    end for
  end for

  for $k = 1, \ldots, endtime$ **do**
    for $i = 1, \ldots, N_p$ **do**
      (2) Prediction Step
      Propagate the particles $\mathbf{x}_k^{(i)}$ using eq. (1).
      Generate measurements $\mathbf{z}_k^{(i)}$ using eq. (4).

      (3) Measurement Update
      Calculate the measurement likelihood:
      $p(\mathbf{z}_k|\mathbf{x}_k^{(i)})$ where $\mathbf{z}_k \sim (0, \mathbf{R}_k)$.
      Update the weights $w_k^{(i)}$ using eq. (11).
    end for
    Normalize the weights: $\hat{w}_k^{(i)} = w_k^{(i)} / \sum_{i=1}^{N_p} w_k^{(i)}$.
    The posterior PDF: $p(\mathbf{x}_k|\mathbf{z}_{1:k}) = \sum_{i=1}^{N_p} \hat{w}_k^{(i)} \hat{\mathbf{x}}_k^{(i)}$.

    (4) Output: The estimated state:
    $\hat{\mathbf{x}}_k = \mathbb{E}[\mathbf{x}_k|\mathbf{z}_{1:k}] \approx \sum_{i=1}^{N_p} \hat{w}_k^{(i)} \hat{\mathbf{x}}_k^{(i)}$.
    (5) Resampling Step
    Set the threshold sample size: $N_{thresh} = N_p/10$.
    Calculate the effective sample size:
    $N_{eff} = 1/\sum_{i=1}^{N_p} (\hat{w}_k^{(i)})^2$
    if $N_{eff} < N_{thresh}$
      The metropolis resampling method is applied.
    end if
  end for

reducing computational complexity. It is a standard practice in particle filtering when the transition model is an adequate approximation of the true state dynamics, allowing reliance on the measurement likelihood in the weighting process, as shown in Eq. (11). The weights are then normalized such that [see Eq. (12)]:

$$\sum_{i=1}^{N_p} w_k^{(i)} = 1. \tag{12}$$

Finally, the resulting posterior PDF is expressed as Eq. (13):

$$p(\mathbf{x}_k|\mathbf{z}_{1:k}) \approx \sum_{i=1}^{N_p} w_k^{(i)} \delta(\mathbf{x}_k - \mathbf{x}_k^{(i)}). \tag{13}$$

The PF algorithm, outlined in Algorithm 1, provides an iterative framework for estimating the state of a dynamic system. In each iteration, the particle states $\mathbf{x_k}$ (position, velocity, acceleration) are first predicted via the motion model in Eq. (1), then the RSS measurements $\mathbf{z_k}$ are collected at the mobile device and converted into distance estimates using the log-normal path-loss model using Eq. (4). These distance estimates are then fused through triangulation to establish the relationship between the measured ranges and the predicted states x_k during the weight-update step.

While this PF framework performs well under ideal conditions, its performance can degrade significantly in real-world scenarios. This is primarily due to the presence of outliers in sensor measurements [33]. Outliers, which may result from measurement noise, interference, or other unexpected phenomena, can skew the estimated state by disproportionately affecting the particle weights. This is because the PF relies on the likelihood function to update the weights of the particles, and outliers can lead to incorrect weight updates, resulting in poor state estimates and increased variance. This sensitivity to outliers is a critical limitation that can undermine the accuracy and robustness of the PF, necessitating additional strategies or alternative approaches are needed to mitigate the impact of such anomalies [34].

B. Outlier-Robust Particle Filter

In this work, to address the outlier problems in sensor measurements, a new outlier-robust particle filter (RPF) is derived. Unlike the PF in Algorithm 1 that models the likelihood PDF with a single noise model, this approach uses two noise models. Let $s \in \{1, 2\}$ denote the noise models, where s_1 corresponds to a multivariate Gaussian distribution, and s_2 corresponds to a multivariate Student's T distribution. The hypothesis indicator $\mathcal{H}_{s,k}$ specifies the selection of the s-th noise model at a specific time step k. The likelihood of transitioning between these noise models is characterized by the transition probabilities, represented as $\mathbf{\Pi}_{s,k}$. This matrix outlines the probability of switching from one noise model to another during a particular time step. Using the transition matrix and the state from the previous time step $k-1$, Eq. (7) can be re-expressed as Eq. (14):

$$p(\mathbf{x}_{k-1}|\mathbf{z}_{1:k-1}) \approx \sum_{i=1}^{N_p} \left(w_{k-1}^{(i)} \delta(\mathbf{x}_{k-1} - \mathbf{x}_{k-1}^{(i)}) \right) \times \sum_{s=1}^{S} \mathbf{\Pi}_{s,k-1}. \tag{14}$$

At a given time step k, assume that the particles are sampled based on a proposal distribution $\mathbf{x}_k^{(i)} \sim q(\mathbf{x}_k|\mathbf{x}_{k-1}^{(i)}, \mathbf{z}_k)$. Under the hypothesis $\mathcal{H}_{k,s}$ where $s \in \{1, 2\}$, the corresponding weights can be calculated recursively as Eq. (15):

$$w_{s,k}^{(i)} \propto w_{k-1}^{(i)} \frac{p_s(\mathbf{z}_k|\hat{\mathbf{x}}_k^{(i)}) p(\hat{\mathbf{x}}_k^{(i)}|\mathbf{x}_{k-1}^{(i)})}{q(\hat{\mathbf{x}}_k^{(i)}|\mathbf{x}_{k-1}^{(i)}, \mathbf{z}_k)}, \tag{15}$$

where, $p_s(\mathbf{z}_k|\hat{\mathbf{x}}_k^{(i)})$ represents the likelihood function corresponding to $\mathcal{H}_{s,k}$, given by Eq. (16):

$$p_s(\mathbf{z}_k|\hat{\mathbf{x}}_k^{(i)}) = \begin{cases} \text{Eq. (5)} & \text{for} \quad \text{s}_1 \\ \text{Eq. (6)} & \text{for} \quad \text{s}_2 \end{cases}. \tag{16}$$

Next, by choosing the proposal important density $q(\hat{\mathbf{x}}_k^{(i)}|\mathbf{x}_{k-1}^{(i)}, \mathbf{z}_k) = p(\hat{\mathbf{x}}_k^{(i)}|\mathbf{x}_{k-1}^{(i)})$, the weights simplifies to Eq. (17):

$$w_{s,k}^{(i)} \propto w_{k-1}^{(i)} p_s(\mathbf{z}_k|\hat{\mathbf{x}}_k^{(i)}), \tag{17}$$

and normalized by Eq. (18):

$$\hat{w}_{s,k}^{(i)} = w_{s,k}^{(i)} / \sum_{i=1}^{N_p} w_{s,k}^{(i)}. \tag{18}$$

Algorithm 2 The Outlier-Robust Particle Filter (RPF).

(1) Input: Initialization

Set the measurement noise model transition probability: $\mathbf{\Pi}_{s,0}$ for $s \in \{1, 2\}$.

for $k = 0$ **do**
  for $i = 1, \ldots, N_p$ **do**
    Samples: $\{\mathbf{x}_0^{(i)} \sim q(\mathbf{x}_0)\}$.
    Set initial weights: $w_0^{(i)} = 1/N_p$.
  end for
end for

for $k = 1, \ldots, endtime$ **do**
  for $i = 1, \ldots, N_p$ **do**
    (2) Prediction Step
    Propagate the particles $\mathbf{x}_k^{(i)}$ using eq. (1).
    Generate measurements $\mathbf{z}_k^{(i)}$ using eq. (4).

    (3) Measurement Update
    Calculate the measurement likelihood:
    $p_s(\mathbf{z}_k|\mathbf{x}_k^{(i)})$ where
    $\mathbf{z}_k \sim \begin{cases} \mathcal{N}(0, \mathbf{R}_k), & \text{for } s_1. \\ St(\mu, \hat{\mathbf{\Sigma}}, v), & \text{for } s_2. \end{cases}$
    Calculate the noise models marginal likelihood
    $p_s(\mathbf{z}_k|\mathbf{z}_{k-1}) = w_{k-1}^{(i)} \times p_s(\mathbf{z}_k|\hat{\mathbf{x}}_k^{(i)})$ for
    $s \in \{1, 2\}$.
    Update the weights $w_{s,k}^{(i)}$ for $s \in \{1, 2\}$
    using eq. (11).
  end for
  Normalize the weights: $\hat{w}_k^{(i)} = w_k^{(i)} / \sum_{i=1}^{N_p} w_k^{(i)}$.
  Noise transition probability:
  $\mathbf{\Pi}_{s,k} = \frac{\mathbf{\Pi}_{s,k-1}^{\varphi}}{\sum_{s=1}^{S} \mathbf{\Pi}_{s,k-1}^{\varphi}}$ for $s \in \{1, 2\}$.
  Update the noise transition probability:
  $\hat{\mathbf{\Pi}}_{s,k} = \frac{\mathbf{\Pi}_{s,k} \times p_s(\mathbf{z}_k|\mathbf{z}_{k-1})}{\sum_{s=1}^{S} \mathbf{\Pi}_{s,k} \times p_s(\mathbf{z}_k|\mathbf{z}_{k-1})}$ for $s \in \{1, 2\}$.
  The posterior PDF:
  $p(\mathbf{x}_k|\mathbf{z}_{1:k}) = \sum_{i=1}^{N_p} \hat{w}_{s,k}^{(i)} \cdot \delta\left(\mathbf{x}_k^{(i)} - \hat{\mathbf{x}}_{s,k}^{(i)}\right) \cdot \sum_{s=1}^{S} \mathbf{\Pi}_{s,k}.$

  (4) Output: The estimated state:
  $\hat{\mathbf{x}}_k = \mathbb{E}[\mathbf{x}_k|\mathbf{z}_{1:k}] = \sum_{s=1}^{S} \sum_{i=1}^{N_p} \hat{w}_{s,k}^{(i)} \cdot \hat{\mathbf{x}}_{s,k}^{(i)} \cdot \mathbf{\Pi}_{s,k}.$

  (5) Resampling Step
  Set the threshold sample size: $N_{thresh} = N_p/10$.
  Calculate the effective sample size:
  $N_{eff} = 1/\sum_{i=1}^{N_p} (\hat{w}_k^{(i)})^2$
  if $N_{eff} < N_{thresh}$
    The metropolis resampling method is applied.
  end if
end for

The transition probability of the $\mathcal{H}_{s,k-1}$ is given by Eq. (19):

$$\mathbf{\Pi}_{s,k-1} = \frac{\mathbf{\Pi}^{\varphi}_{s,k-1}}{\sum_{s=1}^{S} \mathbf{\Pi}^{\varphi}_{s,k-1}}, \tag{19}$$

where, $\varphi \in [0, 1]$ represents the forgetting factor. Utilizing Bayes' theorem, the transition probability for $\mathcal{H}_{s,k}$ is calculated recursively as

$$\mathbf{\Pi}_{s,k} = \frac{\pi^{\varphi}_{s,k-1}\ p_s(\mathbf{z}_k|\mathbf{z}_{k-1})}{\sum_{s=1}^{S} \mathbf{\Pi}^{\varphi}_{s,k-1}\ p_s(\mathbf{z}_k|\mathbf{z}_{k-1})}. \tag{20}$$

Eq. (20) provides an update to each hypothesis probability by blending the prior belief (through the forgetting factor φ) with the marginal likelihood of the new measurement. If the RSS observation aligns closely with Gaussian assumptions, the Gaussian hypothesis is reinforced; if it exhibits a large deviation, the Student's-T hypothesis gains prominence. $p_s(\mathbf{z}_k|\mathbf{z}_{k-1})$ represents the marginal likelihood for $\mathcal{H}_{s,k}$, where the integration is performed over the entire state space of the system described as

$$p_s(z_k|z_{k-1}) = \int_{x_k \in \mathbb{R}^n} p_s(z_k|x_k)\, p(x_k|z_{k-1})\, dx_k. \tag{21}$$

Since $p(\mathbf{x}_k|\mathbf{z}_{k-1}) \approx \sum_{i=1}^{N_p} w^{(i)}_{k-1} \delta_{\hat{\mathbf{x}}^{(i)}_k}$, the integral in Eq. (21) is approximated by Eq. (22):

$$p_s(\mathbf{z}_k|\mathbf{z}_{k-1}) \approx \sum_{i=1}^{N_p} w^{(i)}_{k-1}\ p_s(\mathbf{z}_k|\hat{\mathbf{x}}^{(i)}_k). \tag{22}$$

Finally, the posterior PDF at time step k is formulated as Eq. (23) and Eq. (24):

$$p(\mathbf{x}_k|\mathbf{z}_{1:k}) = \sum_{s=1}^{S} p_s(\mathbf{x}_{s,k}|\mathbf{z}_k) \times \mathbf{\Pi}_{s,k}, \tag{23}$$

$$= \sum_{s=1}^{S} \sum_{i=1}^{N_p} \left(w^{(i)}_{s,k} \delta_{\hat{\mathbf{x}}^{(i)}_k} \times \mathbf{\Pi}_{s,k} \right). \tag{24}$$

Particle weights under model s are computed via the likelihood $p_s(z_k|z_{k-1})$. For the Student's-T hypothesis, a degrees of freedom, ν parameter of 3 is chosen to provide sufficient robustness against heavy-tailed outliers without incurring excessive computational cost. The filter employs $N_p = 200$ particles; whenever fewer than 20 particles (i.e., 10%) carry significant weight, Metropolis resampling is performed to restore diversity. Algorithm 2 summarises the RPF.

IV. Experiments and Results

This section evaluates the performance of the PF and the RPF using both simulated and real-world sensor data for indoor positioning of mobile users. The focus is on estimating the positions of mobile users from Wi-Fi sensor readings, which are often affected by outliers. The simulated data is generated by adding measurement noise and a specified percentage of outliers to a predefined trajectory, emulating real-world conditions. Real-world data consists of Wi-Fi signal strength measurements collected in an indoor environment, reflecting practical challenges such as multipath propagation and signal occlusion.

TABLE I. Simulation Parameters for PF and RPF

Parameter and Symbol	Value and Unit PF	Value and Unit RPF
Number of Particles, N_p	200	200
Control input, $\mathbf{u}_k$	{(0, 0.07) (0.07, 0) (0, -0.07) (-0.07, -0.07) (-0.07, 0) (-0.07, 0.07)}	{(0, 0.07) (0.07, 0) (0, -0.07) (-0.07, -0.07) (-0.07, 0) (-0.07, 0.07)}
Reference RSS, Z_0	0 dBm	0 dBm
Path loss exponent, β	4.0	4.0
Sampling rate, f_s	1 Hz	1Hz
Discretisation period, T	1	1
Reciprocal of the manoeuvre time, α	0.1	0.1
Forgetting factor, φ	non	0.5
degrees of freedom, ν	non	3
Noise Transition Probability, $\mathbf{\Pi}_{s,k}$	non	$\begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix}$
Process Noise Variance, σ^2	$\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$	$\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$
Measurement Noise Variance, σ^2_k	1	1

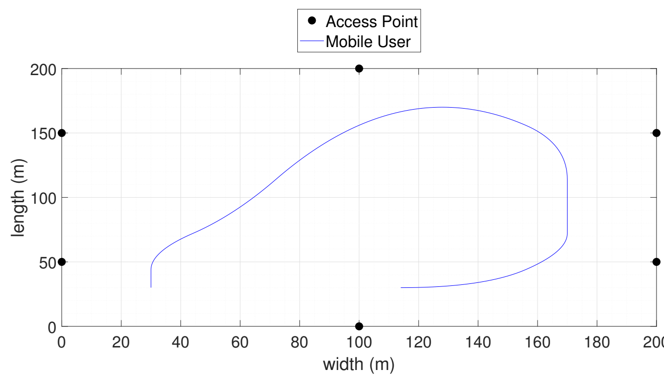

Fig. 1. Mobile user trajectory and access point deployment.

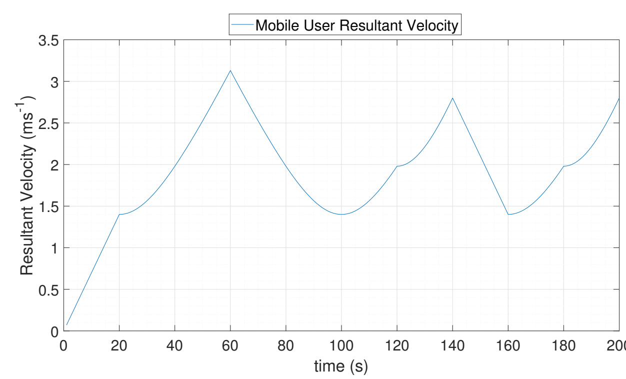

Fig. 2. Mobile user resultant velocity.

A. Simulated Data

Consider a 2D scenario, where a mobile user moves within an indoor building, following the dynamics described by Eq. (1). The user's position is indirectly observed via RSS measurements, modeled by Eq. (4), which are transmitted by the Wi-Fi APs and received by the mobile user's device. Fig. 1 illustrates the mobile user's trajectory alongside the network deployment. The trajectory represents the user's path over 200 time steps, indexed by k, while the network deployment shows the placement of APs within the indoor environment. Fig. 2 shows the resultant speed (magnitude of velocity) of the mobile user at each time step in the simulated scenario. In the simulation, acceleration is set to zero to approximate the user's walking at a constant speed, indicating that each control input $\mathbf{u}_k$ results in constant-speed motion between changes in direction. The network covers an area of 200 m × 200 m, with the APs functioning as transmitter nodes and the mobile user serving as the receiver node. The observed sensor readings correspond to the received Wi-Fi signals, include outliers, that are randomly distributed such that outliers occur in 50% of the time steps, while the remaining 50% of the measurements are unaffected. When outliers are present, the measurements are simulated using a multivariate Student's-T distribution, whereas unaffected data follow a multivariate Gaussian distribution. Here, the control input vector $\mathbf{u}_k$ represents discrete movement commands in the two-dimensional plane, specifically steps to the left, right, up, or down, applied at each time step. The forgetting factor φ =0.5 balances responsiveness to new measurements with retention of past model-selection history, preventing both abrupt and overly sluggish switching between noise models. The simulation parameters are detailed in Table I and the root mean square error (RMSE) is employed as the evaluation metric to assess the accuracy of the PF and the RPF in estimating the position, velocity, and acceleration of the mobile user in the indoor environment.

Fig. 3 compares the estimated trajectories of the mobile user within the network, obtained through the PF and RPF algorithms, based on a single simulation run. It shows that the RPF consistently tracks the mobile user's path more accurately than the PF in most of the time steps. Notably, during periods when the mobile user makes turns, such as between time steps $k = 160$ to $k = 170$, the RPF achieves lower tracking error compared to the PF, as measured by the RMSE.

In the subsequent experiments, the simulation is repeated 100 times under different combinations of noise and outlier distributions and the average RMSE is computed to assess tracking performance. Fig. 4 illustrates the mean RMSE for position estimation for both the PF and RPF algorithms. It is observed that the RPF algorithm outperforms the PF algorithm, achieving a substantially lower RMSE. Specifically, the mean RMSE for the RPF is 9.5208 meters, whereas for the PF it is 15.8734 meters. This results is a significant reduction of approximately 40.02% in RMSE when using the RPF algorithm, highlighting its improved position estimation accuracy as reflected by lower RMSE values.

Fig. 5 illustrates the mean RMSE for velocity estimation using both the PF and RPF algorithms. The results indicate that the RPF algorithm consistently outperforms the PF algorithm in velocity estimation. Specifically, the mean RMSE for velocity using the RPF is 0.1252 ms^{-1}, which is significantly lower than the 0.2035 ms^{-1} observed with the PF algorithm. This represents a reduction of approximately 38.48% in RMSE when using the RPF, indicating improved velocity estimation accuracy, with a lower RMSE compared to the standard method.

Finally, Fig. 6 presents the mean RMSE for acceleration estimation using both the PF and RPF algorithms. It is shows that the RPF consistently achieves lower RMSE for acceleration throughout the simulation time steps. Specifically, the mean RMSE for acceleration with the RPF is $5.7339 \times 10^{-5} ms^{-2}$,

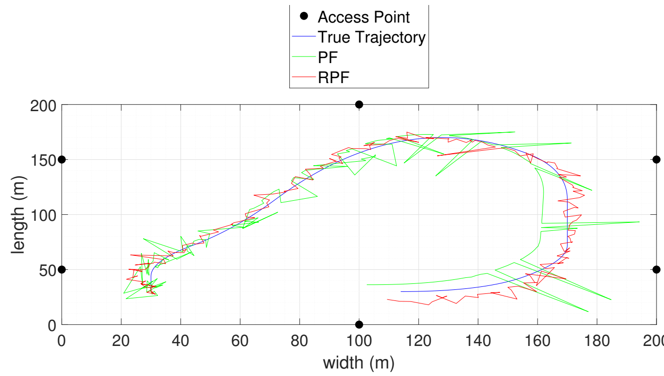

Fig. 3. Estimated trajectory by Particle Filter (PF) and Robust Particle Filter (RPF).

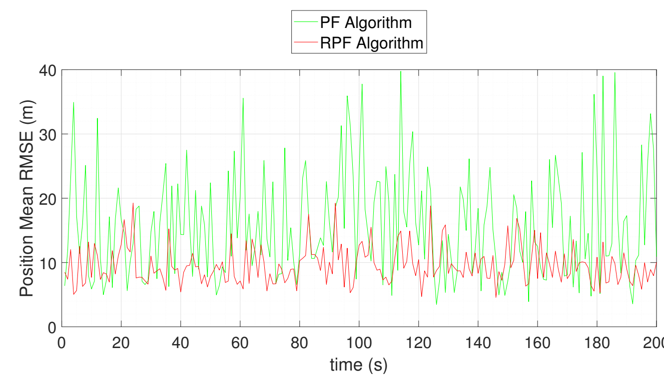

Fig. 4. Mean RMSE of mobile user position using PF and RPF.

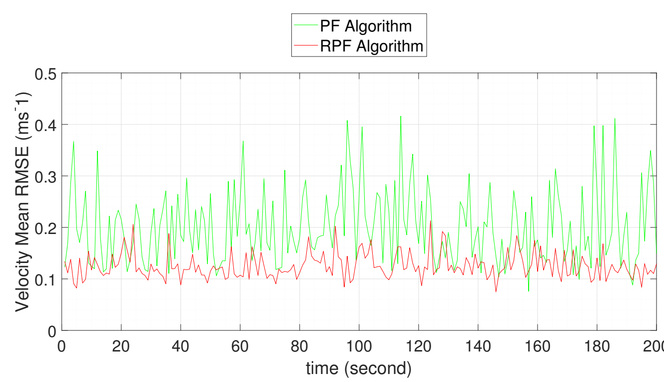

Fig. 5. Mean RMSE of mobile user velocity using PF and RPF.

which is considerably lower than the 1.6767×10^{-4} ms^{-2} observed with the PF. This corresponds to a reduction of approximately 65.80% in RMSE when using the RPF, highlighting its improved accuracy in estimating the acceleration of a mobile user, as indicated by a lower acceleration estimation error.

Overall, the results demonstrate that the RPF outperforms the PF in estimating position, velocity, and acceleration of

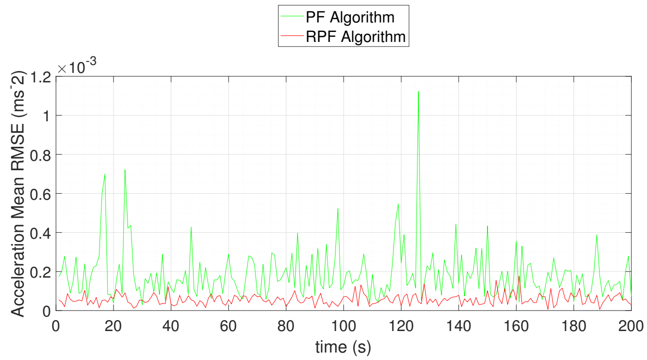

Fig. 6. Mean RMSE of mobile user acceleration using PF and RPF.

a mobile user in indoor environment. The RPF consistently achieves lower RMSE values across all metrics, indicating significant improvements over the PF. This suggests that outliers in RSS measurements can diminish the accuracy of the PF, while the RPF remains robust to such disturbances. As a result, the RPF is more effective at handling outliers in the sensor measurements compared to the PF. Runtime measurements on MATLAB show that a single simulation run takes approximately 0.24 seconds for the PF and 0.33 seconds for the RPF, highlighting the additional computational cost associated with the dual-noise likelihood evaluation. The presence of outliers in highly obstructed and complex indoor environments is expected to significantly degrade the position estimation accuracy of both filters. However, in such outlier-affected scenarios, the RPF consistently outperforms the PF due to its more robust dual-noise likelihood evaluation strategy.

B. Real-World Data

The real-world data was collected at the Melinsung Summer Bay Resort Apartment in Papar, Sabah, a three-room apartment with a total area of 680 square feet. The experimental setup employed the ESP-32-WROOM-32 module, shown in Fig. 7, which served as both the AP and the receiver point (RP). AP and RPs are placed at ground level in the indoor setup (height = 0 m). The parameters z_0 and β in Eq. (4) vary depending on the environment [35], thus the experimental setup, as shown in Fig. 8, was designed to interpolate these parameter values. RPs were positioned at distances ranging from 0.5 to 7.0 meters from the APs. To account for temporal variations and ensure accuracy, measurements were repeated multiple times at different times of the day and under varying environmental conditions. The RSS values were captured and recorded in real-time using the Arduino IDE's serial monitor [36].

To streamline data collection, the PLX-DAQ version 2.11 [37] was utilized, enabling efficient transfer of the readings directly from the serial monitor to an Excel file. The RSS values were collected by connecting the ESP-32-WROOM-32 to a laptop, allowing for real-time data acquisition. The collected RSS measurements visualized in Fig. 9 illustrates the relationship between signal strength and distance. A logarithmic interpolation was applied to identify the optimal fit for the RSS measurements by reducing the squared discrepancies between the observed data points and the corresponding values on the curve. As a result, the parameter z_0 was estimated at -68.4694 dBm, and the path loss exponent β was calculated to be 0.71349, indicating how the RSS decreases logarithmically as the distance increases, given by in both the PF and RPF as a reliable representation of the signal propagation characteristics within the indoor environment.

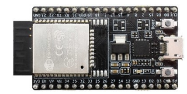

Fig. 7. ESP32-WROOM-32 as Access Point (AP) and Receiver Point (RP) nodes.

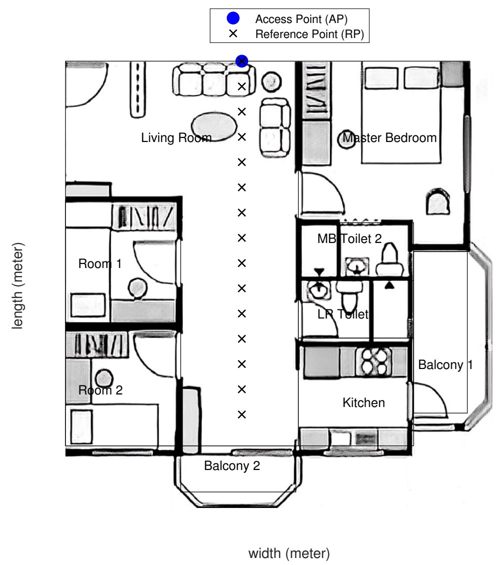

Fig. 8. Transmitter node and receiver node locations for collecting calibrated data.

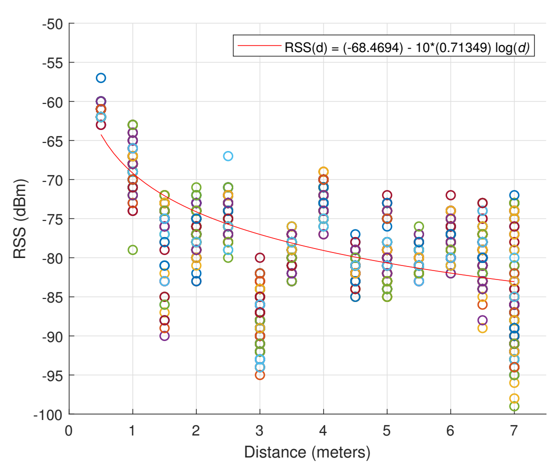

Fig. 9. An interpolated log-normal path loss model derived from calibrated data.

$$Z = -68.4694 - 10 * (0.71349) \log(d) + v, \quad (25)$$

where, v is modeled using a zero-mean Gaussian distribution. It provides the interpolated measurement model, which is used

Next, actual Wi-Fi signal measurements were collected from a mobile user using the experimental setup shown in Fig. 10, which shows the indoor floor plan with four strategically placed Wi-Fi APs and the single 17-second user trajectory from one representative trial. The experimental results are derived from 20 independent real-world trials, each following the same 17-second mobile user trajectory but yielding distinct RSS readings (including natural outliers), and the final outcomes are based on the mean measurements to ensure robustness and reliability of the positioning performance evaluation. As the receiver traversed various locations, the user's trajectory was mapped while capturing RSS values from the Wi-Fi APs. The collected RSS measurements along this trajectory was used to evaluate the performance of the PF and RPF algorithms in accurately estimating the user's position within the indoor environment. During the prediction stage of both algorithms, the measurement model formulated in Eq. (25) was applied.

Fig. 11 presents a comparison of the mean RMSE between the PF and RPF, computed over 17 time steps from 20 repeated experiments using collected Wi-Fi RSS measurements. The data demonstrate that the RMSE of the RPF is consistently lower than that of the PF across all time steps. Notably, the PF shows a significant spike in RMSE at the 13^{th} time step, reaching a value of 43.89825, likely due to outlier measurements or non-Gaussian noise, whereas the RPF remains robust with an RMSE of 0.91450 at the same time step. Overall, the mean RMSE of the PF is 10.27197, while that of the RPF is 0.65683, the difference in RMSE between PF and RPF is 9.61515 reflecting a 93.61% improvement in RPF, highlighting that the RPF has successfully mitigated the effect of outliers and overcome the limitations of the PF.

V. Conclusion

In this study, the performance of the Particle Filter (PF) and the proposed outlier-robust particle filter (RPF) algorithms was thoroughly evaluated using both simulated and real-world data. The results from the simulated data demonstrated that the RPF consistently outperforms the PF in terms of position, velocity, and acceleration estimation. The RPF achieved markedly reduced root mean square error (RMSE) values, with reductions of approximately 40.02% in position, 38.48% in velocity, and 65.80% in acceleration compared to the PF. The real-world experiments further confirmed the robustness of the RPF. By applying a calibrated log-normal path loss model to the collected Wi-Fi RSS data, the RPF outperformed the PF in position estimation accuracy. Notably, the RPF showed a 93.61% improvement in positioning accuracy, with a significant reduction in RMSE at each time step. These results highlight the RPF's robustness in maintaining accurate position estimates, even in the presence of measurement noise and outliers. Therefore the proposed RPF offers a robust solution for tracking and positioning in environments where outliers or noise are prevalent, making it a valuable approach for real-world positioning applications. Future work includes evaluation under more complex indoor scenarios such as multiple trajectories, varied AP deployments, and obstructed layouts with dynamic obstacles, along with integration of map constraints and multi-sensor fusion (e.g., UWB/AoA) to improve positioning robustness in highly non-linear, cluttered environments.

Acknowledgment

The authors express their gratitude to the Ministry of Higher Education Malaysia for providing funding through the Fundamental Research Grant Scheme (FRGS) under project number FRGS/1/2022/TK07/UMS/02/9. Appreciation is also extended to Universiti Malaysia Sabah for their invaluable support in facilitating this research.

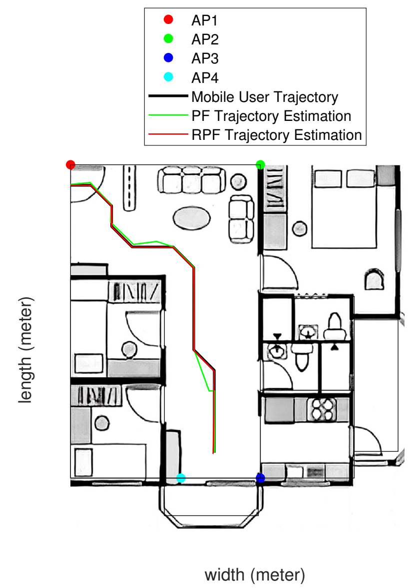

Fig. 10. Indoor floor plan layout, transmitter node and mobile user trajectory for a single simulation run.

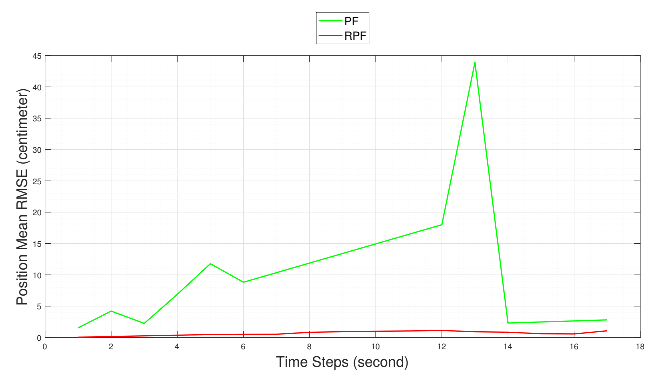

Fig. 11. Comparison of PF and RPF mean position RMSE using collected Wi-Fi RSS measurements.

References

[1] H. Zhang, B. Li, M. Karimi, S. Saydam, and M. Hassan, "Recent advancements in IoT implementation for environmental, safety, and production monitoring in underground mines," *IEEE Internet Things J.*, vol. 10, no. 16, pp. 14 507–14 526, 2023.

[2] J. M. Choi, K. Kim, and S. C. Kim, "Outdoor localization based on RSS ranging aided by pedestrian dead reckoning in GPS restricted scenario," in *Proc. Symp. IEEE VTS Asia Pacific Wireless Commun.*, 2022, pp. 186–190.

[3] J. Singh, N. Tyagi, S. Singh, F. Ali, and D. Kwak, “A systematic review of contemporary indoor positioning systems: Taxonomy, techniques, and algorithms,” *IEEE Internet Things J.*, vol. 11, no. 21, pp. 34 717–34 733, 2024.

[4] L. Bai, F. Ciravegna, R. Bond, and M. Mulvenna, “A low cost indoor positioning system using bluetooth low energy,” *IEEE Access*, vol. 8, pp. 136 858–136 871, 2020.

[5] M. Bhutani, B. Lall, and M. Agrawal, “Optical wireless communications: Research challenges for mac layer,” *IEEE Access*, vol. 10, pp. 126 969–126 989, 2022.

[6] N. Lu and Z. Gao, “Ultra wideband indoor positioning method based on Kalman filter and Taylor algorithm,” in *Proc. 4th Int. Conf. on Intell. Control, Meas. and Signal Process. (ICMSP)*, 2022, pp. 910–914.

[7] R. Kusche, S. O. Schmidt, and H. Hellbrück, “Indoor positioning via artificial magnetic fields,” *IEEE Trans. Instrum. Meas.*, vol. 70, pp. 1–9, 2021.

[8] S. Sadowski, P. Spachos, and K. N. Plataniotis, “Memoryless techniques and wireless technologies for indoor localization with the internet of things,” *IEEE Internet Things J.*, vol. 7, no. 11, pp. 10 996–11 005, 2020.

[9] X. Zhang and W. Sun, “Design of indoor localization algorithm based on Wifi fingerprints,” in *Proc. China Automation Congress (CAC)*, 2021, pp. 6209–6214.

[10] B. Shin, J. H. Lee, C. Yu, H. Kyung, and T. Lee, “Received signal strength-based robust positioning system in corridor environment,” *IEEE Trans. Instrum. Meas.*, vol. 71, pp. 1–15, 2022.

[11] J. A. López-Pastor, M. Poveda-García, A. Gil-Martínez, D. Cañete-Rebenaque, and J. L. Gómez-Tornero, “2-d localization system for mobile iot devices using a single wi-fi access point with a passive frequency-scanned antenna,” *IEEE Internet of Things Journal*, vol. 10, no. 17, pp. 14 995–15 011, 2023.

[12] Q. Zhu, K. Niu, C. Dong, and Y. Wang, “A novel angle of arrival (AOA) positioning algorithm aided by location reliability prior information,” in *Proc. IEEE Wireless Comm. and Netw. Conf. (WCNC)*, 2021, pp. 1–6.

[13] Y. Li, Z. He, Z. Gao, Y. Zhuang, C. Shi, and N. El-Sheimy, “Toward robust crowdsourcing-based localization: A fingerprinting accuracy indicator enhanced wireless/magnetic/inertial integration approach,” *IEEE Internet Things J.*, vol. 6, no. 2, pp. 3585–3600, 2019.

[14] N. H. H. Pham, M. A. Nguyen, and C.-C. Sun, “Indoor positioning system using UWB and Kalman filter to increase the accuracy of the localization system,” in *Proc. IEEE Int. Conf. on Consumer Electron. - Taiwan*, 2022, pp. 339–340.

[15] A. Eman and H. Ramdane, “Mobile robot localization using extended Kalman filter,” in *Proc. 3rd Int. Conf. on Comput. Applications & Inf. Security (ICCAIS)*, 2020, pp. 1–5.

[16] X. Xu, A. Peng, and X. Hong, “Unscented Kalman filtering based multipath-assisted positioning with peak flow tracking,” in *Proc. IEEE 12th Int. Conf. on Indoor Positioning and Indoor Navigation (IPIN)*, 2022, pp. 1–8.

[17] L. Mihaylova, D. Angelova, S. Honary, D. R. Bull, C. N. Canagarajah, and B. Ristic, “Mobility tracking in cellular networks using particle filtering,” *IEEE Trans. Wireless Commun.*, vol. 6, no. 10, pp. 3589–3599, 2007.

[18] H. Gu, H. Mahmoud, R. L. Arun, J. Liu, and X. Ma, “Particle filter and its variants for degradation state estimation and remaining useful life prediction,” in *Proc. Conf. Prognostics and Health Management (PHM)*, 2023, pp. 256–263.

[19] L. Yang, N. Wu, B. Li, W. Yuan, and L. Hanzo, “Indoor localization based on factor graphs: A unified framework,” *IEEE Internet Things J.*, vol. 10, no. 5, pp. 4353–4366, 2023.

[20] Z. Yu, L. Lin, C. Xue, L. Zhuo, and J. Yuan, “Application of untracked particle filter (pso-upf) based on particle swarm optimization in indoor target tracking,” in *Proc. 20th Int. Conf. on Natural Comput., Fuzzy Syst. and Knowledge Discovery (ICNC-FSKD)*, 2024, pp. 1–6.

[21] H. Chen, X. Shen, Y. Wang, and N. Shi, “Particle filter-based enhanced transition model in signal for unsupervised localization,” *IEEE Sensors Journal*, vol. 24, no. 21, pp. 35 845–35 857, 2024.

[22] Y. Shen, B. Hwang, and J. P. Jeong, “Particle filtering-based indoor positioning system for beacon tag tracking,” *IEEE Access*, vol. 8, pp. 226 445–226 460, 2020.

[23] R. Jurdi, H. Chen, Y. Zhu, B. Loong Ng, N. Dawar, C. Zhang, and J. K.-H. Han, “Whereartthou: A wifi-rtt-based indoor positioning system,” *IEEE Access*, vol. 12, pp. 41 084–41 101, 2024.

[24] A. Alteneiji, U. Ahmad, K. Poon, N. Ali, and N. Almoosa, “Indoor localization in multi-path environment based on aoa with particle filter,” in *Proc. 3rd Int. Conf. on Signal Process. and Inf. Security (ICSPIS)*, 2020, pp. 1–4.

[25] W. Li, X. Liu, and L. Guo, “Robust particle filtering with state transition uncertainties: A student’s t disturbance observer-based approach,” *IEEE Trans. Instrum. Meas.*, vol. 70, pp. 1–13, 2021.

[26] J. Wang, P. Dong, K. Shen, X. Song, and X. Wang, “Distributed consensus student-t filter for sensor networks with heavy-tailed process and measurement noises,” *IEEE Access*, vol. 8, pp. 167 865–167 874, 2020.

[27] W. Na, P. Kun, and L. Yang, “A survey of maneuvering target prediction via interactive multiple model,” in *Proc. 34th Chinese Control and Decision Conf. (CCDC)*, 2022, pp. 6069–6074.

[28] J. Zhang, S. Liu, and R. Zhu, “Motion velocity, acceleration and energy expenditure estimation using micro flow sensor,” *IEEE Access*, vol. 7, pp. 75 901–75 909, 2019.

[29] Y. Onykiienko, P. Popovych, R. Yaroshenko, A. Mitsukova, A. Beldyagina, and Y. Makarenko, “Using rssi data for lora network path loss modeling,” in *Proc. IEEE 41st Int. Conf. on Electron. and Nanotechnology (ELNANO)*, 2022, pp. 576–580.

[30] E. I. Adegoke, E. Kampert, and M. D. Higgins, “Empirical indoor path loss models at 3.5Ghz for 5G communications network planning,” in *Proc. Int. Conf. on UK-China Emerging Technol. (UCET)*, 2020, pp. 1–4.

[31] F. A. Abed, Z. A. Hamza, and M. F. Mosleh, “Indoor positioning system based on Wi-fi and bluetooth low energy,” in *Proc. 8th Int. Eng. Conf. on Sust. Technol. and Development*, 2022, pp. 136–141.

[32] A. Kiring, “Shrinkage based particle filters for tracking in wireless sensor networks with correlated sparse measurements,” *PhD thesis, University of Sheffield*, 2017.

[33] A. H. Chughtai, M. Tahir, and M. Uppal, “A robust bayesian approach for online filtering in the presence of contaminated observations,” *IEEE Tras. Instrum. Meas.*, vol. 70, pp. 1–15, 2021.

[34] N. Zhou, L. Lau, R. Bai, and T. Moore, “Evolutionary-inspired strategy for particle distribution optimization in auxiliary particle filtering algorithm based indoor positioning,” in *Proc. Int. Conf. on Indoor Positioning and Indoor Navigation (IPIN)*, 2021, pp. 1–8.

[35] M. A. Mohamed Ghazali, A. Kiring, L. Mihaylova, H. T. Yew, S. K. Chung, and Y. Y. Farm, “Evaluation of theoretical and empirical log-normal models for wi-fi fingerprint mapping in indoor positioning,” in *Proc. IEEE Int. Conf. on Artificial Intell. in Eng. and Technol.*, 2024, pp. 697–702.

[36] K. B, V. P, V. P, N. K, and K. Sriharipriya, “Iot based neonatal patient monitoring system,” in *Proc. 2nd Int. Conf. on Vision Towards Emerging Trends in Commun. and Netw. Technol. (ViTECoN)*, 2023, pp. 1–6.

[37] O. Chidolue and T. Iqbal, “System monitoring and data logging using plx-daq for solar-powered oil well pumping,” in *Proc. 13th IEEE Annual Comput. and Commun. Workshop and Conf.*, 2023, pp. 0690–0694.