RESEARCH ARTICLE

Wireless Channel Estimation for Low-Power IoT Devices Using Real-Time Data

SAMRAH ARIF [ID][1], (Graduate Student Member, IEEE), M. ARIF KHAN [ID][2], (Member, IEEE), AND SABIH UR REHMAN [ID][1], (Member, IEEE)

[1]School of Computing, Mathematics and Engineering, Charles Sturt University, Port Macquarie, NSW 2444, Australia
[2]School of Computing, Mathematics and Engineering, Charles Sturt University, Wagga Wagga, NSW 2650, Australia

Corresponding author: Samrah Arif (sarif@csu.edu.au)

ABSTRACT The Internet of Things (IoT) is gaining immense popularity in executing automation activities via wireless connectivity in the modern era. The IoT networks are designed using mostly low-power IoT (LP-IoT) devices that are battery-operated and have limited computational power. The wireless communication amongst these LP-IoT devices is affected due to the undesirable factors affecting the wireless channel, such as physical obstructions, the distance between devices, wireless network interference, and power limitations of IoT devices. These factors result in attenuation, distortion and phase-shift of the signals arriving at the receiver device. To encounter the effects of the factors affecting the wireless channel in LP-IoT communication, we estimate the wireless channel at the transmitter device before transmission. An effective channel estimation guarantees reliable transmission, improves the throughput rate, and extends the life of the entire IoT network. This study presents two models relevant to LP-IoT communication in IoT networks. The first model is the LP-IoT communication model, which provides a theoretical representation of the wireless channel for the LP-IoT network. The second model is the channel estimation model, where we apply the Least Squares (LSE) and Maximum Likelihood (MLE) techniques to estimate the LP-IoT wireless channel. We analyse the squared error obtained through the LSE and minimise it to reach a Target Error Threshold (TET), where the estimation results are considered accurate. We developed a novel outlier removal method (OUT-R) to eliminate outliers in LP-IoT wireless channel data to achieve this. After outlier removal, we implement the Kalman Filter (KF) method to further improve the channel estimation accuracy. The observation data needed for this investigation has been obtained from real-time measurements in a controlled Line of Sight (LoS) indoor setting using LP-IoT devices. The findings of this study indicate that the suggested method may meet the specified error threshold TET, yielding accurate channel estimation for LP-IoT communication in IoT networks.

INDEX TERMS Wireless channel estimation, low-power IoT devices, RSSI estimation, waspmote, least squares, Kalman filter.

I. INTRODUCTION

The Internet of Things (IoT) has reached a level of enormous potential that qualifies it as a future-enabling technology. Researchers have defined IoT as a network of devices (known as things) connecting people and things via the Internet by incorporating device-specific information such as identification, intelligence, sensing and acting capabilities [1].

The associate editor coordinating the review of this manuscript and approving it for publication was Guangjie Han [ID].

IoT has undeniably revolutionised civilisation through its automated connectivity and ability to assist individuals and industries in automating daily tasks. Smart cities, smart homes, public safety and environment monitoring, smart healthcare, industrial operations, and agriculture are among the most notable IoT applications [2]. A recent study in [3] predicted that by 2030, there would be 29 billion IoT devices worldwide, up from 9.7 billion in 2020. The sensing devices used in IoT communication often have low power, minimal storage, and limited processing power. These limitations
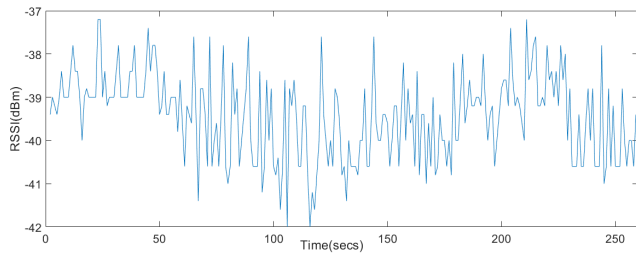
FIGURE 1. Channel behaviour of an actual implemented LP-IoT communication scenario.

impose restrictions on the network architecture, making it difficult for IP networking technologies to communicate reliably. Due to these limitations, IoT devices are often known as constrained devices, and therefore, their communication process is distinguished from cellular or mobile communication [4].

The LP-IoT devices communicate under certain low-power wireless protocols such as Bluetooth, Sigfox, ZigBee and LoRaWAN for data transmission [5]. Efficient and reliable wireless communication amongst LP-IoT devices is critical in any environment. After all, unreliable communication might cause an interruption in automation tasks performed by IoT systems [6]. A vital element for reliable communication is the wireless channel through which the data is transferred from one node (LP-IoT device) to another.

The wireless signal propagating through the wireless channel varies with time due to environmental factors such as objects, walls, floors, electromagnetic interference, and external noise in an indoor environment. These factors influence the wireless signal, which causes reflection, diffraction, scattering, and refraction [7]. These circumstances hamper communication reliability and data rate. Figure 1 illustrates the wireless channel's general behaviour representing the channel's randomness. To maintain optimal communication in LP-IoT devices, the wireless channel parameters must be modelled and estimated before transmission to predict the current channel condition. Such parameters are received signal strength indicator (RSSI), signal-to-noise ratio (SNR), packet delivery ratio (PDR), and link quality indicator (LQI) [8]. The RSSI [9], [10] is a fundamental metric utilised to evaluate the wireless channel for LP-IoT communication. It measures received power by the receiving node (LP-IoT) device. It can be described as "a metric for power received by a radio frequency node device from a sender node or access point". Most LP-IoT devices contain the RSSI register, which indicates the signal strength of the received packet. It can be estimated using certain parameters of LP-IoT wireless channel, such as the time of sent and received packets, the distance between LP-IoT devices, the Time of Signal Arrival (ToA), and the Angle of Arrival (AoA) [11].

Researchers have investigated many estimation techniques to estimate the wireless channel for cellular and IoT communication, such as Least Squares Estimation (LSE) [12], Maximum Likelihood Estimation (MLE) [13], Minimum

Mean Squared Error Estimation (MMSE) [14], and Kalman Filter (KF) method [15]. Each technique has different assumptions and requirements. The LSE method is used to estimate the parameters of a linear regression model [16]. In contrast, the MLE method finds the parameter values that maximise the likelihood function, which is the probability of the observed data [17]. Alternatively, the MMSE approach minimises the Mean Squared Error (MSE) between the actual and predicted data [18].

Similarly, another efficient technique for channel estimation is KF, which estimates the unknown variables based on a series of measurements obtained over a period using two phases: predict and update [19]. Despite the fact that these techniques have positive and negative aspects, we aim to investigate some of them to estimate the wireless channel specifically for LP-IoT devices and minimise the estimation error. Based on the energy restrictions in LP-IoT devices, we selected LSE, MLE, and KF for LP-IoT channel estimation. This paper utilises real-time data to estimate the wireless channel specifically for LP-IoT devices using the LSE, MLE, and KF estimation techniques. Also, this research develops an outlier removal method combined with KF to improve the estimation accuracy. The channel metric this research utilises is RSSI, considered as a time-series LP-IoT channel data. For the data collection, the low-power communication protocol utilised in this research is Zigbee [20] because it is widely used in LP-IoT indoor communications. Thus, this paper consists of two models: the LP-IoT wireless communication model and the LP-IoT wireless channel estimation model. The first model describes the theoretical representation of the LP-IoT wireless channel. The second model describes our implemented method to estimate the LP-IoT wireless channel. Following the estimation techniques LSE and MLE implementation, we minimise the squared error acquired from these methods using the outlier removal (OUT-R) and KF methods. Key contributions of this study are listed below:

- Presented analytical derivation of key parameters for channel estimation techniques (LSE and MLE) for the LP-IoT channel estimation (see Sections IV-B1 and IV-B2).
- Developed a new outlier detection and removal method OUT-R for channel estimation and applied Kalman Filter to minimise estimation error (see Sections IV-B4 and IV-B5).
- Designed an experimental test-bed to collect real-time data for channel characteristics using Waspmote (LP-IoT device) in Line of Sight (LoS) indoor conditions (see Section V).

The remainder of the study is partitioned into the following sections: Section II illustrates the study that has been conducted related to this research. Section III depicts the estimation techniques utilised in this research. Section IV describes the communication and channel estimation models adopted for this study. Section V explains the experimental setup designed to collect real-time data for implementing the

designed system model. Section VI illustrates the results and discussion about this investigation study. Finally, Section VII presents the conclusion of this study and outlines some future work directions.

II. RELATED WORK

This section presents the synopsis of recent studies on wireless channel characterisation for LP-IoT communication. A study conducted in [7] evaluated thirty-four path loss models for wireless IoT technologies based on their characteristics and performance conditions. This study concluded with suggestions for future research areas on wireless channel modelling and estimation.

To further examine the behaviour of wireless IoT technologies, several articles have been written on RSSI in various ways. For instance, the authors in [9] compared the performance of RSSI in different wireless IoT technologies and discussed its challenges and limitations for IoT localisation. In another paper [21], the authors investigated the effect of obstacles on RSSI in wireless sensor networks and proposed solutions to mitigate the impact on localisation and tracking. Alternatively, the authors in [22] presented a bounded-error estimation method based on RSSI and CSI for estimating the distance between wireless sensor devices. In contrast, the authors of [23] proposed an intelligent ZigBee technique to intensify IoT networks' energy efficiency and link quality.

Building on the significance of RSSI in wireless IoT technologies, it is crucial to examine conventional methods for RSSI estimation, such as Least Squares Estimation (LSE). The LSE method has played a fundamental role in improving the efficiency and reliability of IoT systems by reducing the estimation error. One of its applications can be seen in [24], where the authors developed an adaptive RSSI-based ranging system for IoT networks in an outdoor environment. The efficacy of the proposed system was analysed and evaluated under various conditions, including the effect of environmental conditions such as humidity and temperature. The results of experiments and simulations demonstrated the accuracy and reliability of the designed scheme, concluding that it offers improved performance compared to traditional approaches and is a promising solution for outdoor wireless sensor networks. In another study [25] based on the LSE method for channel estimation in NB-IoT downlink systems, the authors have mitigated the channel estimation errors incurred by the standard LSE method without requiring additional frequency-band resources. Through comprehensive simulations, authors have demonstrated the superiority of the proposed model to conventional LSE; however, despite emphasising its low complexity, the stringent computational complexity analysis still needs improvement. In addition, the algorithm's applicability to various NB-IoT scenarios must be strengthened further. Also, the absence of the data sets restricts opportunities for further investigation. In [26], the use of RSSI data for modelling the path loss in a Long Range (LoRa) network was investigated. In this paper, the authors

presented a method for collecting and analysing the RSSI data in a LoRa network and used this data to develop a path loss model. The detailed outcomes of the experiments and simulations demonstrated the proposed model's validity, accuracy, and reliability. The authors also discussed the implications of the proposed model for the design and deployment of LoRa networks.

One of the effective uses of RSSI is estimating the location and tracking objects in wireless sensor networks. Many researchers have used the KF method for estimating locations and tracking such as [27] and [28]. In [27], the author designed a system that uses the RSSI values to estimate the location of objects in the network and track their movements over time using the KF. The results of experiments and simulations are presented to assess the practicability and precision of the designed method. In [28], the KF method was used to establish a link quality estimation technique. In this paper, the author demonstrated that wireless networks are susceptible to mobility, making predicting the occurrence and stability of linked and transitional areas more challenging. The author estimated link quality fluctuation by combining the RSSI values of arriving (ACK) packets with the Packet Delivery Ratio (PDR) using a KF to encounter this issue. The KF method improved the accuracy of the predictions by 12%.

Similarly, the authors in [29] presented more findings on reducing the estimation error in IoT channel estimation by proposing a group sparsity estimation method. This method exploits the sparsity pattern in data transmissions while permitting concurrent detection of active devices and estimates their channels. This approach shortens the duration of the required signature sequence length, improving IoT access efficiency. Furthermore, the authors have presented a smoothing method to efficiently manage high-dimensional structured estimations while balancing estimation precision and computational cost. However, in practice, the implementation may be complicated by environmental factors and device diversity. Moreover, the practical applicability of this study may also be affected by computational costs, particularly in large IoT networks.

Currently, there is a significant interest among scientists pertaining to the integration of Machine Learning (ML) approaches in the domain of channel estimation as evidenced by various studies [30], [31], [32], [33]. However, it is essential to acknowledge the persistent benefits of traditional approaches, characterised by their simplicity and transparency, as they are built on well-established relationships and assumptions. This transparency enables a more straightforward process of interpretation and troubleshooting, which is a crucial feature in the intricate field of channel estimation. On top of that, conventional approaches demonstrate practicality in situations where limited data is available since they require fewer data points for reasonable forecasts. This attribute is particularly desirable in practical contexts where obtaining substantial ML training data might be arduous. Additionally, conventional techniques

exhibit stability and robustness, ensuring reliable channel estimates despite fluctuations in input data, hence reducing vulnerability to outliers and noise.

These findings contribute to the comprehension of wireless channel behaviour. However, a research gap emerges when directly estimating wireless channels using only RSSI without incorporating independent features. These limitations are due to the lack of practical applicability in real-world environments, inadequate datasets for reproducible results, and high estimation errors [33]. Given the existing gaps in research, in this paper, we propose a model to estimate the wireless channel with RSSI as a primary metric using various existing estimation techniques. Additionally, we have developed a new outlier removal method, particularly for reducing the estimation error of wireless channels in LP-IoT devices. The outcome of our model exhibits much lower MSE than the MSE reported in [33] for various datasets.

III. PRELIMINARIES
This section presents the preliminary concepts utilised in the design of this study. These include LSE, MLE, outlier removal, and KF method. These techniques are often used for estimating wireless channels; thus, we aim to implement these techniques for LP-IoT wireless channels.

A. LEAST SQUARES ESTIMATION (LSE)
The LSE method is one of the basics of the estimation theory [34]. It is used for identifying an unknown parameter or signal from a measurement or observation in the presence of uncertainty or noise. It is an approach seen in regression analysis to get the Best Fit Line (BFL) that accurately portrays the correlation between an independent and dependent variable. The BFL is stretched across a scatter plot of data points to reflect the relationship between those data points. This estimation method is among the most successful for determining the BFL. Further explanation and implementation of LSE for LP-IoT channel estimation can be found in Section IV-B1.

B. MAXIMUM LIKELIHOOD ESTIMATION (MLE)
The MLE is a statistical technique for calculating the optimal mean or standard deviation of the distribution of given data. It maximises the log-likelihood function to estimate the channel parameters and compares model fits using information theory. MLE's objectives are to discover the best estimate of unknown parameter values based on observations and evaluate different model parameters to pick the model that fits the channel data the best [35].

C. OUTLIERS REMOVAL
The outliers are the number of observations that deviate greatly from most data points. They can result from measurement variability, data entry errors, or extreme values not typical of the data set. The removal of these outliers can significantly improve the estimation accuracy. Literature mentions numerous methods for detecting and removing

TABLE 1. Symbols and notations from Section IV-A to IV-B2.

Notation	Description
$\mathcal{D}_t$	Set of LP-IoT transmitter devices
$\mathcal{D}_r$	Set of LP-IoT receiver devices
d_m	m^{th} LP-IoT transmitter device
d_n	n^{th} LP-IoT receiver device
y	Received signal
$\mathbf{x}$	Transmitted signal vector
h	Wireless channel vector
v	Thermal noise
α, β	Real Gaussian random variables
p	Probability density function
$p_\alpha(a)$	Probability density function of α
$p_\beta(b)$	Probability density function of β
A	Amplitude of channel h
ϕ	Phase of channel h
σ^2	Variance
γ	Measured RSSI (mean of five readings at each time instant
$\hat{\gamma}$	Estimated RSSI
ϵ	Estimation error
S_ϵ	Sum of squared error
w	Number of estimation parameters
ρ_0	Estimated y-intercept of best fit line
ρ_1	Esitimated slope of best fit line
l	Receiver location
K	Number of data samples
τ	Time (seconds)
$f(\epsilon_k)$	PDF of the estimation error
$\mathcal{L}$	Likelihood function

outliers, such as the Z-Score method, the Interquartile Range (IQR) method, and Tukey's fences method [36]. In this study, we utilise IQR and Tukey's fences technique to detect and eliminate the outliers in the measured data.

D. KALMAN FILTER (KF)
The KF is an algorithm that uses measurements and predictions to estimate the state of a system over time. It is utilised in various applications, including navigation, control systems, and signal processing, where there is uncertainty in the data. The main principle underlying the KF is to use a combination of predictions and measurements to estimate the system's true state, considering the uncertainty in both measurements and predictions. The algorithm updates its estimate of the state over time as new measurements become available, leading to a more accurate estimate of the true state of the system. The KF provides an advantage in that it is light on memory because it does not need to keep any history other than the previous state. This quality suits real-time estimation for wireless sensor networks, especially LP-IoT applications, which can not tolerate complex calculations [37].

IV. SYSTEM MODEL
This section presents two models: the first reflects the theoretical model of the LP-IoT communication for decentralised network architecture, while the second depicts the implementation of certain channel estimation approaches along with the process to reduce estimation error in the presented IoT communication model. Readers are referred to Table 1 to understand the symbols and notations used in this
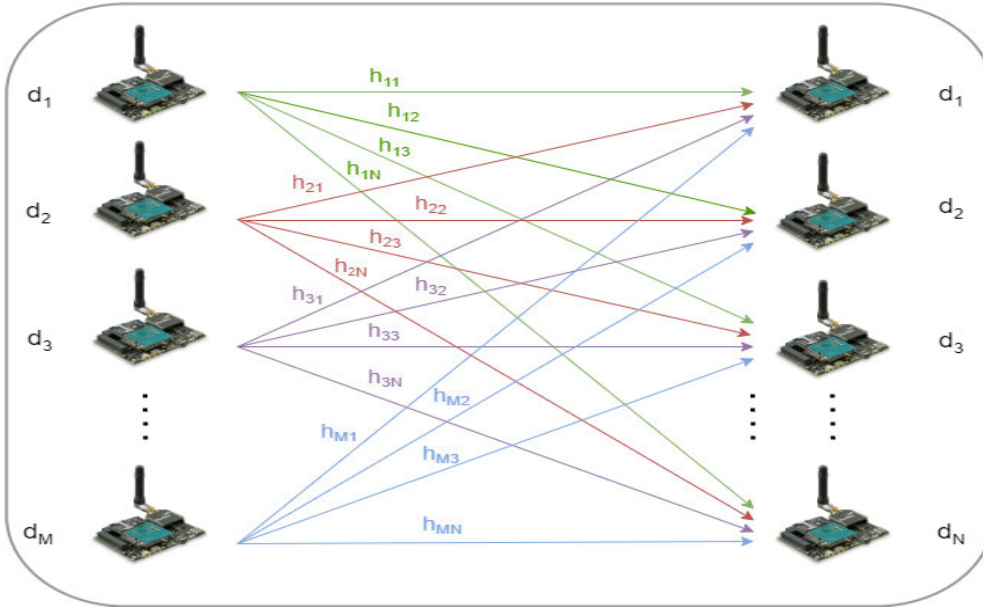
FIGURE 2. Generalised model for the communication of LP-IoT devices with *M* transmitters and *N* receivers.
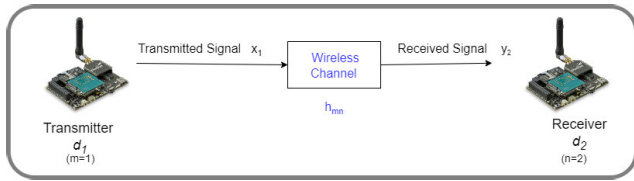

FIGURE 3. An experimental implementation schematic of two IoT devices (LoS Communication).

study from Section IV-A to Section IV-B2. Table 2 defines the rest of the symbols and notations.

A. LP-IoT WIRELESS COMMUNICATION MODEL

Let us assume an LP-IoT network where there are M number of total transmitter LP-IoT devices such that $\mathcal{D}_t = \{d_1, d_2, d_3, \cdots, d_M\}$ and N number of total receiver LP-IoT devices such that $\mathcal{D}_r = \{d_1, d_2, d_3, \cdots, d_N\}$ in the network. The transmitting and receiving nodes have a Line of Sight (LoS) communication within $3m$ communication range as shown in Figure 2. Let us consider m as any transmitter from the set of transmitter devices $\mathcal{D}_t$, and n as any receiver from the set of receiver devices $\mathcal{D}_r$ that are communicating at any time instant. The transmitter device d_m sends the signal vector $\mathbf{x}$ to the receiver d_n at the current time step through channel h_{mn}. The signal received at device d_n can be written as:

$$y_n = h_{mn}\mathbf{x} + v_n, \qquad (1)$$

where $\mathbf{x}_{(M \times 1)}$, and v_n represent the transmitted signal and the thermal noise respectively. Furthermore, the thermal noise v_n at device d_n can be modelled as a Gaussian random variable.

Now, let us define the wireless channel between the LP-IoT device pair (d_m, d_n) as a complex number $h_{mn(1 \times N)}$ which can

mathematically be written as:

$$h_{mn} = \alpha + j\beta, \qquad (2)$$

where both α and β are Gaussian random variables with each having $\mathcal{N}(0, 1)$ distribution. Since both α and β are assumed as real Gaussian random variables (RVs), using [38] we can write PDF of α and β as

$$p_\alpha(a) = \frac{1}{\sqrt{2\pi\sigma_\alpha^2}} \exp\left[-\frac{(a - \overline{\alpha})^2}{2\sigma_\alpha^2}\right], \qquad (3)$$

and the pdf of β can be written as:

$$p_\beta(b) = \frac{1}{\sqrt{2\pi\sigma_\beta^2}} \exp\left[-\frac{(b - \overline{\beta})^2}{2\sigma_\beta^2}\right], \qquad (4)$$

where $\overline{\alpha} \triangleq E\{\alpha\}$ and $\overline{\beta} \triangleq E\{\beta\}$ are the statistical means, whereas σ_α^2 and σ_β^2 are the variances of α and β respectively. We assume that all the channels in the system are fading channels that fade independently. By assuming the wireless channel as a complex random variable, it can be represented by 'Euler's formula' [39] in terms of its amplitude A and phase ϕ as:

$$h_{mn} = A_{mn}e^{j\phi_{mn}},$$

where A_{mn} and ϕ_{mn} represent the variations in the amplitude and the direction of the signal between the device m and n respectively, and $e^{j\phi_{mn}}$ can be represented as $(cos\phi_{mn} + jsin\phi_{mn})$. The h_{mn} can now be represented as:

$$h_{mn} = A_{mn}[cos(\phi_{mn}) + jsin(\phi_{mn})],$$

where A_{mn} is multiplied to $cos(\phi_{mn})$ and $jsin(\phi_{mn})$ and is represented as:

$$h_{mn} = A_{mn}cos(\phi_{mn}) + jA_{mn}sin(\phi_{mn}),$$

which leads to the same representation of channel parameters as defined in equation 2. The magnitude of the wireless channel between two LP-IoT devices, denoted by h_{mn} can be represented as:

$$|h_{mn}| = \sqrt{(\alpha)^2 + (\beta)^2}, \qquad (5)$$

whereas the phase ϕ_{mn} between transmitter and the receiver device is given as:

$$\phi_{mn} = \tan^{-1}\left(\frac{\beta}{\alpha}\right). \qquad (6)$$

In an ideal condition, the magnitude of h_{mn} should equal 1, which means the received signal is the same as the transmitted signal without creating any distortion in the received signal, i.e. $y = x$. However, in the practical scenario, h_{mn} is a time-variant complex-valued number estimated at the transmitter before transmitting the signal. Assuming the noise to be 0 i.e. $v_n = 0$, the received signal y_n with the estimated channel $\hat{h}_{mn}$ between the two LP-IoT nodes can be represented as:

$$y_n = \hat{h}_{mn}\mathbf{x}. \qquad (7)$$

Overall, the primary goal of this study is to estimate the LP-IoT channel h_{mn} between d_m and d_n for the next time steps using the current time steps. A specific scenario that represents the communication between d_m and d_n is illustrated in Figure 3, where m and n denotes any transmitter and receiver from the set of transmitter devices $\mathcal{D}_t$ and receiver devices $\mathcal{D}_r$ respectively. To estimate the LP-IoT wireless channel, we consider RSSI as an LP-IoT channel metric collected by the practical deployment of LP-IoT devices. To estimate RSSI, we investigate techniques that minimise the sum of squared error (SSE) between the actual and predicted LP-IoT channel values. A detailed explanation of the process is described in the subsequent subsection.

B. LP-IoT WIRELESS CHANNEL ESTIMATION MODEL

In this section, we provide the model that uses several estimation techniques to estimate the LP-IoT RSSI-based wireless channel. First, the LSE and MLE were implemented, and their squared errors were obtained. These squared errors were employed to establish the target error threshold (TET) at which the implemented technique offers the most accurate estimation. The (OUT-R) outlier removal method was devised and applied to satisfy the TET value. The maximum squared error (via the LSE method) of the preserved channel values after the OUT-R method was compared with TET. The highest squared error measured by OUT-R did not correspond to the TET. Then, we applied the KF method to the measured channel data and compared its maximum squared error to TET, which was near the TET value. Lastly, we combined the Out-Filter method with KF. For this, we utilised the retained channel values from the OUT-R method to implement KF and compared its maximum squared error to the TET. The combination of OUT-R and KF demonstrated that the best estimation could be obtained by combining these two

methods. While initially designed for a specific scenario, this method can be implemented within a distributed IoT network, as depicted in figure 2. Notably, its implementation is viable due to its minimal requirements in terms of execution time and memory resources. For implementing this model, we use the real RSSI-based LP-IoT channel data acquired from the practical implementation in an indoor environment. Figure 4 illustrates the estimation model that highlights the process of this designed model.

1) LEAST SQUARES ESTIMATION (LSE) FOR LP-IoT WIRELESS CHANNEL

Consider the LP-IoT channel estimation problem where γ represents the RSSI-based LP-IoT channel. The RSSI is one of the wireless channel metrics utilised in wireless channel assessment. Thus, we aim to estimate RSSI as a time-series observation by the LSE method. The basic formulation of the LSE approach consists of a set of K pairs of observations (τ_k, γ_k) that are used to determine a function that relates the value of the dependent variable γ to the values of the independent variable τ. we consider only one dependent variable γ, with a linear function. The general equation of the estimated BFL using the LSE method is given as:

$$\hat{\gamma} = \rho_0 + \rho_1 \tau, \qquad (8)$$

where $\hat{\gamma}$ denotes the variable representing the estimated LP-IoT channel values, τ denotes the variable which is time-series in this case, ρ_0 denotes the y-intercept and ρ_1 denotes the slope of the BFL. We assume that there are some estimation errors between the measured and the estimated channel values, which can be represented mathematically in a generalised form as:

$$\epsilon = \gamma - \hat{\gamma}, \qquad (9)$$

which can also be represented as:

$$\gamma = \rho_0 + \rho_1 \tau + \epsilon, \qquad (10)$$

The variable ϵ represents the estimation error such as $\epsilon \neq 0$. Ideally, the estimation error ϵ should equal 0. However, in practical implementation, there is always some estimation error, which can be either positive or negative. The positive error raises the estimated value to the actual value, which needs to be lower. Similarly, the negative error decreases the estimated value to the value that must be risen to get closer to the actual value, thus decreasing the estimation error. The ρ_0 and ρ_1 are called coefficients of regression that specify the estimation parameters, which are fed into the LSE model equation to get the BFL. Therefore, we must estimate the parameters ρ_0 and ρ_1. The acquired BFL minimises the sum of the squared errors between the actual and the estimated LP-IoT channel values. The sum of squared error can be
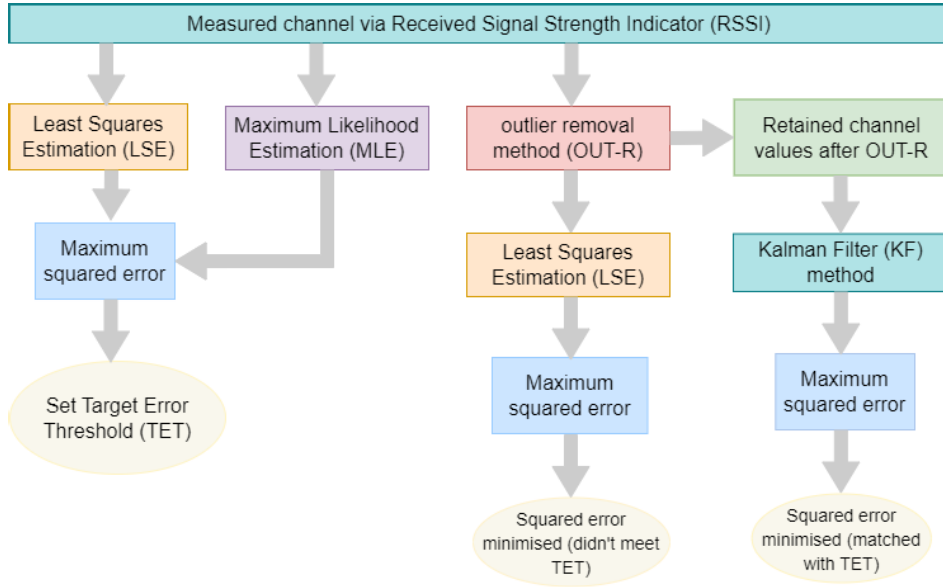
FIGURE 4. A LP-IoT wireless channel estimation model incorporates a number of estimating methods.

represented mathematically as [40]:

$$S_\epsilon = \sum_{k=1}^{K} (\gamma_k - \hat{\gamma}_k)^2$$

$$\implies \sum_{k=1}^{K} (\gamma_k - (\rho_0 + \rho_1 \tau_k))^2, \tag{11}$$

where S_ϵ denotes the sum of squared error between the actual and estimated data, which needs to be minimised. This sum of squared error S_ϵ represents the closeness between the actual and the predicted channel values [40]. To reduce the sum of squared error, we take the derivative of S_ϵ with respect to ρ_0 and ρ_1 and setting them to zero gives the estimated values of parameters ρ_0 and ρ_1 of BFL equation. To calculate ρ_0, we write the derivative of estimation error with respect to ρ_0 as:

$$\frac{\partial S_\epsilon}{\partial \rho_0} = \sum_{k=1}^{K} \frac{\partial S_\epsilon}{\partial \rho_0} (\gamma_k - (\rho_0 + \rho_1 \tau_k))^2.$$

To get the partial derivative with respect to ρ_o, we apply the power rule and chain rule to get:

$$\frac{\partial S_\epsilon}{\partial \rho_0} = -2 \sum_{k=1}^{K} \frac{\partial S_\epsilon}{\partial \rho_0} (\gamma_k - (\rho_0 + \rho_1 \tau_k)),$$

which can also be written as:

$$\frac{\partial S_\epsilon}{\partial \rho_0} = 2K\rho_0 + 2\rho_1 \sum_{k=1}^{K} \tau_k - 2 \sum_{k=1}^{K} \gamma_k.$$

By setting the partial derivative of S_ϵ with respect to ρ_o equal to 0, and solving the equation gives the estimation

parameter ρ_0 (y-intercept) of BFL as:

$$2K\rho_0 + 2\rho_1 \sum_{k=1}^{K} \tau_k - 2 \sum_{k=1}^{K} \gamma_k = 0,$$

$$\rho_0 = \frac{1}{K} \left(\sum_{k=1}^{K} \gamma_k - \rho_1 \sum_{k=1}^{K} \tau_k \right),$$

which can also be represented as:

$$\rho_0 = \bar{\gamma}_k - \rho_1 \bar{\tau}_k, \tag{12}$$

where $\bar{\gamma}_k$ denotes the mean of measured RSSI values, and $\bar{\tau}_k$ denotes the mean time.

Next, we calculate ρ_1 by taking the derivative of estimation error with respect to ρ_1 and set them to 0 as:

$$\frac{\partial S_\epsilon}{\partial \rho_1} = 2\rho_1 \sum_{k=1}^{K} (\tau_k)^2 + 2\rho_0 \sum_{k=1}^{K} \tau_k - 2 \sum_{k=1}^{K} \gamma_k \tau_k,$$

$$2\rho_1 \sum_{k=1}^{K} (\tau_k)^2 + 2\rho_0 \sum_{k=1}^{K} \tau_k - 2 \sum_{k=1}^{K} \gamma_k \tau_k = 0,$$

which is then solved to get the second estimation parameter ρ_1 as a slope of BFL, and can be represented mathematically as:

$$\rho_1 = \frac{\sum_{k=1}^{K} (\gamma_k - (\frac{1}{K} \sum_{k=1}^{K} \gamma_k))(\tau_k - (\frac{1}{K} \sum_{k=1}^{K} \tau_k)))}{\sum_{k=1}^{K} (\tau_k - (\frac{1}{K} \sum_{k=1}^{K} \tau_k))^2},$$

which can also be written as:

$$\rho_1 = \frac{\sum_{k=1}^{K} (\gamma_k - \bar{\gamma}_k)(\tau_k - \bar{\tau}_k)}{\sum_{k=1}^{K} (\tau_k - \bar{\tau}_k)^2}. \tag{13}$$

The estimated data points illustrating the estimated LP-IoT channel at each time instant can be formed using the LSE

equation as:

$$\hat{\gamma}_k = \rho_0 + \rho_1 \tau_k + \epsilon_k, \tag{14}$$

where $\hat{\gamma}_k$ denotes the variable representing the observed LP-IoT channel values at k^{th} instant, and τ_k denotes the variable which is time-series in this case. The greater the deviation between these data points and the anticipated BFL, the more error is added. Thus, the LSE technique is founded on the principle that the square of the acquired errors must be as little as feasible. We consider the squared error against each value of γ for the estimation analysis. The estimation error at any time τ_k is defined as:

$$\epsilon_k = \hat{\gamma}_k - \gamma_k, \tag{15}$$

where γ_k denotes the actual acquired RSSI (LP-IoT channel value), and $\hat{\gamma}_k$ represents the estimated RSSI at each time step. To reduce the possibility of inaccuracy due to negative errors, the LSE error equation is represented in the form of squared error as:

$$\epsilon_k^2 = (\hat{\gamma}_k - \gamma_k)^2, \tag{16}$$

which leads to calculating the sum of squared error as shown in equation 11. Next, we aim to estimate RSSI-based wireless channel γ in matrix form. We take the vector parameter ρ of dimension $w \times 1$, representing the estimation parameter ρ_0 and ρ_1. As we tend to estimate the BFL with two estimation parameters, ρ_0 and ρ_1, we take $w = 2$. For the estimated LP-IoT channel $\hat{\gamma} = [\hat{\gamma}_1, \hat{\gamma}_2, \dots, \hat{\gamma}_k]^T$ to be linear in the unknown parameters, the best approximate solution to get the estimated channel values of $\hat{\gamma}$ through LSE using matrix notation is written in closed form as:

$$\hat{\gamma} = X\rho, \tag{17}$$

where X is a known $K \times w$ matrix ($K > w$) of full rank w. The matrix X is referred to as the observation matrix without the noise PDF assumption. The open form of matrix notations for $\hat{\gamma}$, X, and ρ can be written as:

$$\begin{bmatrix} \hat{\gamma}_1 \\ \hat{\gamma}_2 \\ \vdots \\ \hat{\gamma}_k \end{bmatrix} = \begin{bmatrix} \tau_1 & 1 \\ \tau_2 & 1 \\ \vdots & \vdots \\ \tau_k & 1 \end{bmatrix} \begin{bmatrix} \rho_1 \\ \rho_0 \end{bmatrix}. \tag{18}$$

The LSE solution $S_\epsilon(\rho)$ minimises the sum of the squares of the entries of vector $\hat{\gamma} - X\rho$ which represents the error between the actual and estimated values as shown in equation 11. The vector ρ represents the estimator vector. The LSE solution of estimator ρ is found by minimising the sum of squared error $S_\epsilon(\rho)$ in matrix form. The $S_\epsilon(\rho)$ in matrix form is represented as:

$$S_\epsilon(\rho) = \sum_{k=1}^{K} (\gamma_k - \hat{\gamma}_k)^2$$
$$= (\gamma - X\rho)^T (\gamma - X\rho),$$

which is expanded, and solved mathematically to get:

$$S_\epsilon(\rho) = \gamma^T \gamma - \gamma^T X\rho - \rho^T X^T \gamma + \rho^T X^T X\rho. \tag{19}$$

By taking the derivative of equation 19 with respect to ρ, we can write as:

$$\frac{\partial S_\epsilon(\rho)}{\partial \rho} = \frac{\partial}{\partial \rho} (\gamma^T \gamma - \gamma^T X\rho - \rho^T X^T \gamma + \rho^T X^T X\rho). \tag{20}$$

By solving the derivative shown in equation 20, the term $\gamma^T \gamma$ is eliminated because it does not involve ρ. As ρ^T has dimensions $1 \times \omega$ and X^T is a $\omega \times K$ matrix, the multiplication of these matrices gives the matrix having dimension $1 \times K$ which is then multiplied to γ having dimensions $K \times 1$. Therefore, the term $\rho^T X^T \gamma$ is reduced to scalar value and is eliminated from the equation. Now, the equation 20 becomes:

$$\frac{\partial S_\epsilon(\rho)}{\partial \rho} = -2X^T \gamma + 2X^T X\rho.$$

Setting the derivative of $S_\epsilon(\rho)$ equal to zero as:

$$-2X^T \gamma + 2X^T X\rho = 0,$$

which is then solved and gives the LSE estimator as:

$$\rho = (X^T X)^{-1} X^T \gamma, \tag{21}$$

which provides the two estimation parameters ρ_0 and ρ_1 to draw the BFL. The estimator values ρ are fed into equation 17 to get the estimated LP-IoT channel values. Next, another estimation technique MLE is applied to the LP-IoT channel values described in the subsequent subsection.

2) MAXIMUM LIKELIHOOD ESTIMATION (MLE) FOR LP-IoT WIRELESS CHANNEL

We aim to apply MLE using the LSE method by assuming the given vector of LP-IoT wireless channel observations as $\gamma_1, \gamma_2, \dots, \gamma_k$. For this, we consider a previously used LSE model as shown in equation 9. Using the LSE equation, we target to estimate ρ_0 and ρ_1 utilising the MLE method. We assume that all the estimation errors e_k at any time instant are independently identically distributed (IID) and follows the normal distribution as $e_k \sim \mathcal{N}(0, \sigma^2)$. In this model, we have e_k as a random variable; therefore, the probability density function (PDF) of the error e_k can be written as:

$$f(\epsilon_k) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{1}{2\sigma^2}(\epsilon_k - 0)^2}, \tag{22}$$

where $f(\epsilon_k)$ represents the PDF of the estimation error ϵ_k at k^{th} instant. We first need to determine the likelihood function (LF) for MLE. The LF of ϵ_k is the Joint Probability Distribution (JPDF) of all the random variables. As we have assumed that estimation errors in this model are IID having normal distribution, we can write the LF as the product of all PDFs of estimation error as:

$$\mathcal{L}(\epsilon) = \prod_{k=1}^{K} f(\epsilon_k), \tag{23}$$

where $\mathcal{L}(\epsilon)$ represents the LF of all estimation errors. By substituting the equation 22 into equation 23, We can represent LF as:

$$\mathcal{L}(\epsilon) = \prod_{k=1}^{K} \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{1}{2\sigma^2}(\epsilon_k)^2}, \qquad (24)$$

which is simplified by product rule as:

$$\mathcal{L}(\epsilon) = \left(\frac{1}{\sqrt{2\pi\sigma^2}}\right)^K e^{-\frac{1}{2\sigma^2}\sum_{k=1}^{K}(\epsilon_k)^2}.$$

The LF can also be written using the equation 14 as:

$$\mathcal{L}((\rho_0, \rho_1, \sigma^2)|\tau) = \left(\frac{1}{\sqrt{2\pi\sigma^2}}\right)^K e^{-\frac{1}{2\sigma^2}\sum_{k=1}^{K}(\gamma_k-\rho_0-\rho_1\tau_k)^2},$$

The objective of MLE is to determine the model parameter values ρ_0 and ρ_1 that maximise the LF over the parameter space. Since we know that the log function is a monotonically increasing function, it is easier to handle the log-likelihood functions rather than dealing with the LF. Therefore, the log-likelihood function of the model is expressed by taking the natural logarithm on both sides as:

$$\ln \mathcal{L}((\rho_0, \rho_1, \sigma^2)|\tau)$$
$$= \ln \left(\frac{1}{\sqrt{2\pi\sigma^2}}\right)^K e^{-\frac{1}{2\sigma^2}\sum_{k=1}^{K}(\gamma_k-\rho_0-\rho_1\tau_k)^2}.$$

Using the properties of logarithms, we can then simplify the expression by splitting the logarithm of the product into the sum of the logarithms of the factors as:

$$\ln \mathcal{L}((\rho_0, \rho_1, \sigma^2)|\tau)$$
$$= K\left\{\ln\left(\frac{1}{\sqrt{2\pi\sigma^2}}\right)\right\} - \frac{1}{2\sigma^2}\sum_{k=1}^{K}(\gamma_k-\rho_0-\rho_1\tau_k)^2. \quad (25)$$

Simplifying the first term of equation 25 by using the logarithm rule to get:

$$\ln \mathcal{L}((\rho_0, \rho_1, \sigma^2)|\tau)$$
$$= -K\left\{\ln\sigma + \ln(\sqrt{2\pi})\right\} - \frac{1}{2\sigma^2}\sum_{k=1}^{K}(\gamma_k-\rho_0-\rho_1\tau_k)^2. \quad (26)$$

By rearranging the terms in equation 26, we can write as:

$$\ln \mathcal{L}((\rho_0, \rho_1, \sigma^2)|\tau)$$
$$= K\left\{-\ln\sigma - \ln(\sqrt{2\pi})\right\} - \frac{1}{2\sigma^2}\sum_{k=1}^{K}(\gamma_k-\rho_0-\rho_1\tau_k)^2. \quad (27)$$

By simplifying the logarithm of the product of the first term in equation 27, the equation of log-likelihood can be written as:

$$\ln(\mathcal{L}(\rho_0, \rho_1, \sigma^2|\tau))$$
$$= -\frac{K}{2}\ln(2\pi) - \frac{K}{2}\ln(\sigma^2) - \frac{1}{2\sigma^2}\sum_{k=1}^{K}(\gamma_k-\rho-\rho_1\tau_k)^2. \quad (28)$$

To find out the maximum likelihood estimators, we obtain the partial derivatives of equation 28 with respect to each of the three parameters ρ_0, ρ_1, and σ^2 as:

$$\frac{\partial}{\partial\rho_0}\ln(\mathcal{L}) = -\frac{1}{\sigma^2}\sum_{k=1}^{K}(\gamma_k-\rho_0-\rho_1\tau_k), \qquad (29)$$

$$\frac{\partial}{\partial\rho_1}\ln(\mathcal{L}) = -\frac{1}{\sigma^2}\sum_{k=1}^{K}(\gamma_k-\rho_0-\rho_1\tau_k)\tau_k, \qquad (30)$$

$$\frac{\partial}{\partial\sigma^2}\ln(\mathcal{L}) = -\frac{K}{2\sigma^2} + \frac{1}{2\sigma^4}\sum_{k=1}^{K}(\gamma_k-\rho_0-\rho_1\tau_k)^2. \quad (31)$$

To maximise the log-likelihood, we set the partial derivatives with respect to ρ_0, ρ_1, and σ^2 equal to 0. Since we are only concerned with the estimation parameters ρ_0 and ρ_1, the equation 29 and 30 will be utilised for the estimation of these parameters. For ρ_0, we can write as:

$$\frac{1}{\sigma^2}\sum_{k=1}^{K}(\gamma_k-\rho_0-\rho_1\tau_k) = 0,$$

which is then solved to get the estimated $\rho0$ as follows:

$$\rho_0 = \bar{\gamma}_k - \rho_1\bar{\tau}_k.$$

Similarly, the estimated ρ_1 is calculated by setting the derivative with respect to $\rho1$ as shown in 32.

$$\frac{1}{\sigma^2}\sum_{k=1}^{K}(\gamma_k-\rho_0-\rho_1\tau_k)\tau_k = 0, \qquad (32)$$

By solving equation 32, we get the estimated ρ_1 which is represented as:

$$\rho_1 = \frac{\sum_{k=1}^{K}(\gamma_k-\bar{\gamma}_k)(\tau_k-\bar{\tau}_k)}{\sum_{k=1}^{K}(\tau_k-\bar{\tau}_k)^2}.$$

3) DETERMINING TARGET ERROR THRESHOLD (TET)

After implementing the linear estimation techniques (LSE and MLE) on RSSI-based LP-IoT channel estimation, we analyse the squared error acquired from LSE estimation. At this stage, we are not considering the squared error acquired from MLE as it gives the approximate same squared error as LSE. There is a need to determine the acceptable squared error limit, referred to as the target error threshold (TET), that gives us the maximum estimation accuracy to minimise the estimation error. For this, we first calculate the confidence interval of LP-IoT channel data to acquire the TET. We assume that the LP-IoT wireless channel follows the Gaussian distribution. Therefore, we set the interval of $\mu \pm 2\sigma$ from the PDF of LP-IoT channel data, which becomes a 95.44% confidence interval. By using equation 33 and 34, we get the confidence interval that lies between $L_{CI} = -39.2108dBm$ and $U_{CI} = -39.0893dBm$. This interval gives the squared error interval between $\epsilon_L = 0.0287dBm$ and
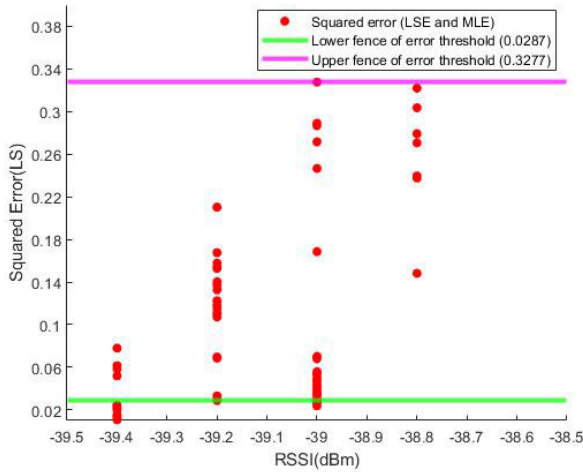
FIGURE 5. Determination of target squared error threshold (TET).

FIGURE 6. A visual representation of major and minor outliers boundaries with the range of retained RSSI after OUT-R method.

$\epsilon_U = 0.3277dBm$ as shown in Figure 5.

$$L_{CI} = \rho_0 + \left\{ 0.9544(\sigma/\sqrt{(K)}) \right\}, \qquad (33)$$

$$U_{CI} = \rho_0 - \left\{ 0.9544(\sigma/\sqrt{(K)}) \right\}, \qquad (34)$$

where L_{CI} represents the lower boundary of confidence interval, U_{CI} represents the upper boundary of confidence interval, ϵ_L denotes the lower limit squared error corresponding to L_{CI}, ϵ_U denotes the upper limit of squared error corresponding to U_{CI}, ρ_0 denotes the estimated y-intercept parameter of BFL, σ denotes the standard deviation of the measured data, and K denotes the number of measurements. We set the upper limit of the squared error, i.e. $0.3277dBm$, as the TET. After identifying the TET, we aim to remove the outliers in the observed data described in the subsequent subsection. The symbols and notations utilised from Section IV-B3 to Section IV-B5 are explained in Table 2.

4) OUT-R METHOD

The next step to reduce the estimation error is to remove the outliers. The general outliers are considered as the number of observations outside the range of the other observations closer to the mean value [41]. In the LP-IoT wireless channel estimation scenario, we need to develop a method of outliers removal that significantly reduces the estimation error and reaches up to the TET, even if the data points are not too far from the mean value. We aim to find the data points γ_k that do not occur frequently. For this, we divide the data into a group of 46 data points. As we have 262 LP-IoT channel samples, we get 5.69 by dividing 262 samples by 46. Adding a constant of 2 becomes 7.69, and rounding off the result provides number 8. This value is the basis for identifying the outlier for LP-IoT channel data. The equation of the outlier detection method can be mathematically expressed as:

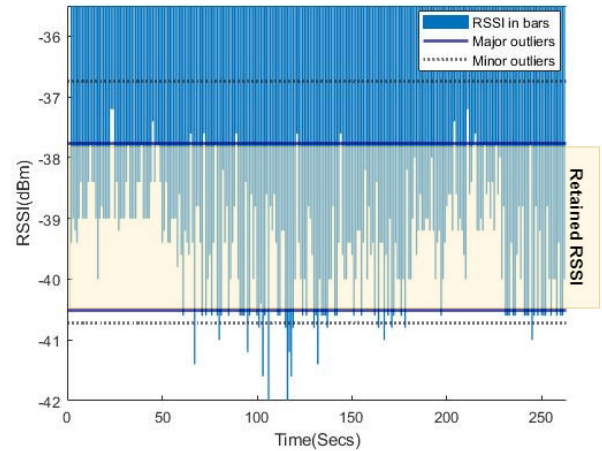$$OutD = \frac{K}{G} + 2, , \qquad (35)$$

where K indicates the number of data samples and G denotes the quantity of data samples in each group. We assume that the γ values at which the number of data points are less than 8, we consider it as an outlier in our case and will remove it. This method retains the 93.5199% of the data points, which will be used further to reduce the estimation error.

Using this method, we detect the retained γ between $-40.8 \ dBm \le \gamma \le -37.6 \ dBm$ which yields 93.5199% data. We select this range of γ values because it excludes γ values at which the number of data points is lower than or equal to 8, calculated using equation 35. The retained values of γ can be achieved by calculating major outliers' upper and lower fences. The major outliers fences can be calculated by using equations 36 and 37 and the minor outliers fences are calculated by using equations 38 and 39 respectively. The minor outliers can also be used to remove the outliers in some cases. However, we do not consider minor outliers at this stage.

$$\mathcal{U}^{mj} = Q3 + \{a(IQR\}, \qquad (36)$$

$$L^{mj} = Q1 - \{b(IQR)\}, \qquad (37)$$

where $Q1$ and $Q3$ denotes the $25th$ and $75th$ percentile, while IQR represents the interquartile range which has been acquired through subtracting $Q1$ from $Q3$. The filter parameter a and b is set to 0.6857 and 0.1425 respectively. For calculating the upper $\mathcal{U}_{mn}$ and lower L_{mn} minor boundaries, the following equations have been formulated:

$$\mathcal{U}^{mr} = Q3 + \{f(IQR)\}, \qquad (38)$$

$$L^{mr} = Q1 - \{g(IQR)\}, \qquad (39)$$

where f and g denotes the filter parameter which has been set to $0.6857 * 2$ and $0.1425 * 2$ respectively. Figure 6 shows the graphical representation of major and minor fences of outliers acquired from equation 36, 37, 38 and 39. Figure 12a shows the LSE implementation after removing the major outliers, significantly improving the BFL and reducing the squared error between the measured and estimated values as shown

TABLE 2. Symbols and notations from Section IV-B3 to IV-B5.

Notation	Description
TET	Target error threshold
L_{CI}	Lower boundary of confidence interval
U_{CI}	Upper boundary of confidence interval
ϵ_L	Lower limit squared error corresponding to L_{CI}
ϵ_U	Upper limit of squared error corresponding to U_{CI}
G	Number of data samples in each group
$Q1$	25th percentile
$Q3$	75th percentile
IQR	Interquartile range
$\mathcal{U}^{mj}$	Upper boundary of major outliers
L^{mj}	Lower boundary of major outliers
$\mathcal{U}^{mr}$	Upper boundary of minor outliers
L^{mr}	Lower boundary of minor outliers
$\mathbf{s}_\tau$	State vector at time τ
$\mathbf{z}_{1:\tau-1}$	Measurement vector from time 1 to $\tau-1$
$\mathbf{P}$	Covariance matrix
$p(\mathbf{s}_\tau|\mathbf{z}_{1:\tau-1})$	Predicted density of state $\mathbf{s}_\tau$ given $\mathbf{z}_{1:\tau-1}$
$p(\mathbf{s}_\tau|\mathbf{z}_{1:\tau})$	Updated posterior density of state $\mathbf{s}_\tau$ given $\mathbf{z}_{1:\tau}$
$\hat{\mathbf{s}}_{\tau|\tau-1}$	Predicted mean state vector
$\mathbf{P}_{\tau|\tau-1}$	Covariance of predicted mean state vector
$\hat{\mathbf{s}}_{\tau|\tau}$	Updated mean state vector
$\mathbf{P}_{\tau|\tau}$	Covariance of updated mean state vector
$\mathbf{A}$	State transition matrix
$\mathbf{q}$	Process noise vector
$\mathbf{H}$	Measurement model matrix
$\mathbf{r}$	Measurement noise vector
$\mathbf{Q}$	Process noise covariance matrix
$\mathbf{v}$	Innovation vector
$\mathbf{S}$	Innovation covariance matrix
$\mathbf{K}$	Kalman gain

in Figure 12b. The minor outliers do not significantly impact the squared error compared to the major outliers. Therefore, we do not consider minor outliers in this case. We are still a bit far away from achieving our TET. As retained data points still do not meet the TET, we investigate the KF method to reduce further squared error and improve the estimation accuracy.

5) KALMAN FILTER (KF) METHOD FOR LP-IoT CHANNEL ESTIMATION

KF is used to estimate the current state of any system based on the previous state. It is an online technique and thus considered to be computationally efficient for state estimation, and hence is suitable for estimation in LP-IoT devices [42]. KF provides an analytical solution for linear and Gaussian models by recursively computing predicted density (in the prediction step) and updated posterior density (in the update step) [37]. In the prediction step, KF computes the prediction step as:

$$p(\mathbf{s}_\tau|\mathbf{z}_{1:\tau-1}) = \mathcal{N}(\mathbf{s}_\tau : \hat{\mathbf{s}}_{\tau|\tau-1}, \mathbf{P}_{\tau|\tau-1}),$$

where $\mathbf{s}_\tau$ is the state vector at time τ, $\mathbf{z}_{1:\tau-1}$ is the measurement vector from time 1 to $\tau - 1$, $p(\mathbf{s}_\tau|\mathbf{z}_{1:\tau-1})$ is the predicted density of the state $\mathbf{s}_\tau$ given the measurements $\mathbf{z}_{1:\tau-1}$. Due to the assumption of linear and Gaussian model for process and measurement, the predicted density is Gaussian centred at the predicted mean state, $\hat{\mathbf{s}}_{\tau|\tau-1}$ and $\mathbf{P}_{\tau|\tau-1}$ as the covariance. In the update step, the posterior

density is updated by taking new measurements into account as:

$$p(\mathbf{s}_\tau|\mathbf{z}_{1:\tau}) = \mathcal{N}(\mathbf{s}_\tau : \hat{\mathbf{s}}_{\tau|\tau}, \mathbf{P}_{\tau|\tau}),$$

where $\mathbf{z}_{1:\tau}$ is the measurement from time 1 to τ (current measurement also included), $p(\mathbf{s}_\tau|\mathbf{z}_{1:\tau})$ is the updated posterior density of the state vector $\mathbf{s}_\tau$ given the measurements vector $\mathbf{z}_{1:\tau}$. The updated density is also Gaussian, centred at updated mean state denoted by $\hat{\mathbf{s}}_{\tau|\tau}$ and covariance denoted by $\mathbf{P}_{\tau|\tau}$. Since the Gaussian distribution is completely defined by its mean and covariance, $\hat{\mathbf{s}}_{\tau|\tau-1}$, $\mathbf{P}_{\tau|\tau-1}$, $\hat{\mathbf{s}}_{\tau|\tau}$ and $\mathbf{P}_{\tau|\tau}$ are only required to be computed in each recursion.

Let us consider the LP-IoT system as a dynamic system that changes its state over a reasonable time. For implementing the update and prediction steps, we take the LP-IoT wireless channel measurements retained after removing the identified outliers by the OUT-R method (explained in Section IV-B4) in the system. The number of data samples retained after the OUT-R method is 246. We aim to estimate the retained channel values (denoted by γ) by the KF method (using the update and prediction step). The LP-IoT channel data is estimated in terms of RSSI measurements. Also, the measurements include some random noise due to the surrounding environment, which affects the wireless channel data. In the experiment scenario, the location l_1 and distance between transmitter d_1 and receiver d_2 (at l_1) are fixed. Figure 7 depicts the prediction and update steps of KF's implementation. The process model equation for the LP-IoT system (considered linear and Gaussian) is defined as:

$$\mathbf{s}_\tau = \mathbf{A}_{\tau-1}\mathbf{s}_{\tau-1} + \mathbf{q}_{\tau-1},$$

where $\mathbf{s}_\tau$ is the state vector (denoted by vector notation, however scalar in two-device communication case) at time τ, representing the estimated RSSI value, $\mathbf{A}_{\tau-1}$ denotes the state transition matrix having dimension 1×1, $\mathbf{s}_{\tau-1}$ denotes the state at the previous time, and $\mathbf{q}_{\tau-1}$ indicates the process noise vector (scalar for two-device communication). In this case, $\mathbf{A}_{\tau-1}$ is the identity matrix and $\mathbf{q}_{\tau-1}$ has the Gaussian distribution such that $\mathbf{q}_{\tau-1} \sim \mathcal{N}(0, \mathbf{Q}_{\tau-1})$. Now, the measurement model equation, which is linear and Gaussian, can be represented as:

$$\mathbf{z}_\tau = \mathbf{H}_\tau\mathbf{s}_\tau + \mathbf{r}_\tau,$$

where, $\mathbf{z}_\tau$ is the measurement at time τ, $\mathbf{H}_\tau$ denotes the measurement model matrix having dimension 1×1, $\mathbf{r}_\tau$ represents the measurement noise vector such that $r_\tau \sim \mathcal{N}(0, \mathbf{R}_\tau)$ and $\mathbf{R}_\tau$ represents the measurement noise covariance matrix. Next, we aim to calculate the predicted mean and covariance for the prediction step, followed by the updated mean and covariance for the update step. The predicted mean, and the predicted covariance for the LP-IoT channel are computed as follows:

$$\hat{\mathbf{s}}_{\tau|\tau-1} = \mathbf{A}_{\tau-1}\hat{\mathbf{s}}_{\tau-1|\tau-1},$$
$$\mathbf{P}_{\tau|\tau-1} = \mathbf{A}_{\tau-1}\mathbf{P}_{\tau-1|\tau-1}\mathbf{A}_{\tau-1}^T + \mathbf{Q}_{\tau-1},$$

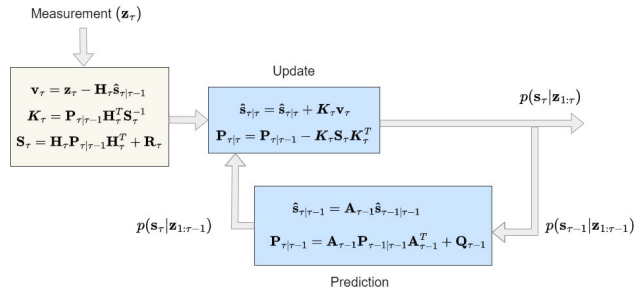**FIGURE 7.** A standardised illustration of the KF technique used to estimate the LP-IoT wireless channel.

TABLE 3. General characteristics of Waspmote platform [43].

Waspmote Platform	Characteristics
Power Consumption	On: 17 mA Sleep: 30 µA Deep Sleep: 33 µA Hibernate: 7 µA
Input/Output	7 analog inputs, 8 digital I/O 2 UARTs, 1 I2C, 1 SPI, 1 USB
Sensors embedded on board	Accelerometer: ±2g/±4g/±8g Low power: 0.5 / 1 / 2 / 5 / 10 Hz Normal mode: 50 / 100 / 400 / 1000 Hz
Electrical characteristics	Battery voltage: 3.3-4.2 V USB charging: 5 V – 100 mA Solar panel load: 6-12 V – 300 mA

where $\mathbf{Q}_{\tau-1}$ represents the process noise covariance matrix, and ^{T} denotes the transpose operation. To calculate the updated mean and covariance, we first need to acquire the innovation at time τ, innovation covariance at time τ, and Kalman gain. The innovation can be defined as the difference between the predicted and actual measurements. This difference may result from various factors, such as measurement noise, modelling errors, and unmodeled disturbances. The innovation is utilised by the KF to update the state estimate and covariance matrix to minimise the prediction error. Specifically, the KF employs the innovation to calculate the Kalman gain matrix, which determines the balance between the prediction and the measurement in the state update step. A large innovation (i.e., a significant discrepancy between the predicted and actual measurement) will lead to a greater Kalman gain, which gives the measurement more weight in the state update step. In contrast, a small innovation will produce a smaller Kalman gain, increasing the prediction's weight during the state update step. The equation for predicting the measurement using the state estimate is written as follows:

$$\hat{\mathbf{z}}_\tau = \mathbf{H}_\tau \hat{\mathbf{s}}_\tau. \tag{40}$$

KF assumes that measurement noise is distributed normally having zero-mean, with a known covariance matrix $\mathbf{R}_\tau$. As a result, the KF predicts the expected measurement value based on the state estimate alone, without considering the measurement noise, as mentioned in equation 40. Utilising the predict measurement equation, the equation of innovation at time τ can be written as:

$$\mathbf{v}_\tau = \mathbf{z}_\tau - \hat{\mathbf{z}}_\tau$$
$$\implies \mathbf{z}_\tau - \mathbf{H}_\tau \hat{\mathbf{s}}_{\tau|\tau-1},$$

where $\mathbf{v}_\tau$ denotes the innovation vector at time τ. The innovation covariance matrix is written as:

$$\mathbf{S}_\tau = \mathbf{H}_\tau \mathbf{P}_{\tau|\tau-1} \mathbf{H}_\tau^T + \mathbf{R}_\tau,$$

where $\mathbf{S}_\tau$ represents the innovation covariance at time τ. Now, using the innovation covariance matrix, the Kalman gain can be determined as follows:

$$\mathbf{K}_\tau = \mathbf{P}_{\tau|\tau-1} \mathbf{H}_\tau^T \mathbf{S}_\tau^{-1},$$

where $\mathbf{K}_\tau$ denotes the Kalman gain at time τ. After calculating the Kalman gain, we aim to calculate the updated mean state and covariance. The updated mean and the updated covariance can be obtained from the following equations, respectively:

$$\hat{\mathbf{s}}_{\tau|\tau} = \hat{\mathbf{s}}_{\tau|\tau-1} + \mathbf{K}_\tau \mathbf{v}_\tau,$$
$$\mathbf{P}_{\tau|\tau} = \mathbf{P}_{\tau|\tau-1} - \mathbf{K}_\tau \mathbf{S}_\tau \mathbf{K}_\tau^T.$$

This implementation is a scalar version of the KF used for communication between two LP-IoT devices. However, vectors and matrices have been used in the equations because the vector and matrix notation facilitates us to convey the equations briefly and efficiently. In the scalar variant of the KF, both the state and the measurements are uni-dimensional, so the vectors and matrices are simplified to scalar values. For example, the state vector $\mathbf{s}$ becomes a scalar value s, whereas the measurement matrix $\mathbf{H}$ becomes a scalar value h.

V. EXPERIMENTAL SETUP
This section describes the required hardware and its setup to conduct this investigation study, along with a detailed explanation of data collection by setting up the scenarios in an indoor CSU laboratory environment.

A. PLATFORM FOR DATA COLLECTION
The experimental test-bed was established utilising Waspmote sensor devices [45]. Waspmote, an IoT sensor device platform, supports multiple sensor technologies, expansion boards, and communication protocols. Its comprehensive and well-documented codebase has shown field-proven reliability and stability. It is currently extensively used in IoT research [46], [47], [48], [49]. The hardware architecture of Waspmote has been specifically developed to operate with minimal power. Table 3 provides technical specifications for a Waspmote device. It explains the detailed power consumption specifications across various operational states, encompassing 'On,' 'Sleep,' 'Deep Sleep,' and 'Hibernate' modes. It also highlights the platform's I/O capabilities, featuring seven analog inputs and eight digital I/O pins, along with essential communication interfaces, including UARTs (Universal Asynchronous Receiver-Transmitter), I2C (Inter-

TABLE 4. XBEE PRO communication parameters [44].

XBEE-PRO S1 Parameters	Physical Values
Total Number of Channels	12
Channel Numbers Used	12 (0x0C) - 23 (0x17)
Channel Width	5 MHz bandwidth per channel
Data Rate	115200 bps
Observed Channel	13 (2.410 – 2.415 GHz)
Protocol	802.15.4
Transmitted Power (dBm)	18dBm (63.1 mW)

Integrated Circuit), SPI (Serial Peripheral Interface), and USB (Universal Serial Bus). Furthermore, it emphasises the onboard sensors, specifically the embedded accelerometer sensor, which can measure multi-axis acceleration across different ranges. It outlines the available sampling rates in low-power and normal modes. Finally, the table mentions the key electrical characteristics, covering battery voltage, USB charging parameters, and solar panel compatibility. This comprehensive information is of great value to researchers seeking to choose an appropriate IoT platform for experimentation, considering the diverse range of IoT devices and their corresponding applications.

Numerous communication technologies are used to communicate between IoT devices. ZigBee is the most widely used technology for short-distance and indoor communication. It is used in most IoT applications that require short-range communication [50] [51]. XBee-Pro S1, a variant of ZigBee, has a centre frequency of 2.4GHz, and it supports the requirements of LP-IoT networks. This radio frequency module has been designed according to the IEEE 802.15.4 standards [52]; hence, it has been used for this preliminary experiment with the Waspmote sensor device. Table 4 demonstrates the parameters for XBEE PRO S1 utilised in this research.

B. CONFIGURATION AND SETUP

The XBee-Pro S1 was configured by XCTU (an open-source multi-platform for powerful wireless network configuration options and architecture) [53] for configuring XBee-Pro S1. The main configuration settings include setting up the operating modes of XBee-Pro S1. The XBee-Pro S1 was configured by XCTU (an open-source multi-platform for powerful wireless network configuration options and architecture) for configuring XBee-Pro S1. The main configuration settings include setting up the operating modes of XBee-Pro S1. XBee can be operated in transparent (AT) and Application Programming Interface (API) modes. The AT mode facilitates serial communication between the transmitter and receiver, and the data is immediately transferred to the identified destination address. The API mode provides the structured interface and is used to communicate with multiple devices at a time by organising the packets in a frame-based structure; hence, it assists in establishing complex communication [54]. At this stage, we set the devices in AT mode as we are considering a basic two-device setup that does not involve

TABLE 5. Five RSSI observations $\gamma_1(\tau)$ to $\gamma_5(\tau)$ and its mean $\gamma(\tau)$ for initial twenty seconds.

$Time(\tau)$ (secs)	$\gamma_1(\tau)$ (dBm)	$\gamma_2(\tau)$ (dBm)	$\gamma_3(\tau)$ (dBm)	$\gamma_4(\tau)$ (dBm)	$\gamma_5(\tau)$ (dBm)	$\gamma(\tau)$ (dBm)
0:01:01	-47	-39	-39	-36	-36	-39.4
0:01:02	-39	-39	-39	-39	-39	-39
0:01:03	-40	-39	-39	-39	-39	-39.2
0:01:04	-39	-40	-40	-39	-39	-39.4
0:01:05	-42	-39	-39	-36	-39	-39
0:01:06	-39	-39	-39	-36	-39	-38.4
0:01:07	-39	-39	-39	-39	-39	-39
0:01:08	-39	-39	-39	-39	-39	-39
0:01:09	-39	-39	-39	-39	-39	-39
0:01:10	-42	-36	-36	-39	-39	-38.4
0:01:11	-36	-39	-39	-39	-36	-37.8
0:01:12	-39	-39	-39	-39	-36	-38.4
0:01:13	-39	-39	-39	-39	-36	-38.4
0:01:14	-39	-39	-39	-39	-39	-39
0:01:15	-47	-39	-39	-39	-36	-40
0:01:16	-39	-39	-39	-39	-39	-39
0:01:17	-39	-40	-40	-36	-39	-38.8
0:01:18	-39	-39	-39	-39	-39	-39
0:01:19	-39	-39	-39	-39	-39	-39
0:01:20	-39	-39	-39	-39	-39	-39

complex communication. It is also required to configure the devices by simultaneously updating the value of the first module's DH and DL parameters to the values of the second module's SH and SL parameters. The channel values based on RSSI over a certain time were collected using the Python libraries.

C. TEST-BED DESIGN

The designed experimental test-bed for this research is based on the basic two-device setup where one Waspmote IoT device acts as a transmitter d_1 while the other acts as a receiving node d_2 having LoS communication between them. Using customised source code, the packet information was received using Waspmote's open-source IDE (Integrated Development Environment). Figure 8 illustrates the lab layout of the experiment in the laboratory. The collected data includes the timestamps, sent packets, transmitter ID, transmitter power level and RSSI represented by (γ) at the receiver. However, we are interested only in RSSI at each timestamp because this data will be used to model h later in this research. The timestamp has been inserted before receiving each packet on the receiver device d_2. The transmit power P_t of Waspmote is $18dBm$. Channel C was used for the ZigBee (XBee-PRO S1) communication.

1) EXPERIMENT SCENARIO - RSSI VS TIME

In this scenario, the Waspmote transmitter d_1 and the Waspmote receiver d_2 were placed at a fixed distance of 2.75m in the CSU research laboratory having LoS communication. The transmitter was placed on the centre table in the room, and the receiver was placed on a chair having different heights of table and chair. Both devices are connected to the laptop. The surrounding environment has computers, chairs, a centre table, and a whiteboard, as shown in Figure 8. We collected five readings of the RSSI till 262 seconds using this setup. Then, we calculated the
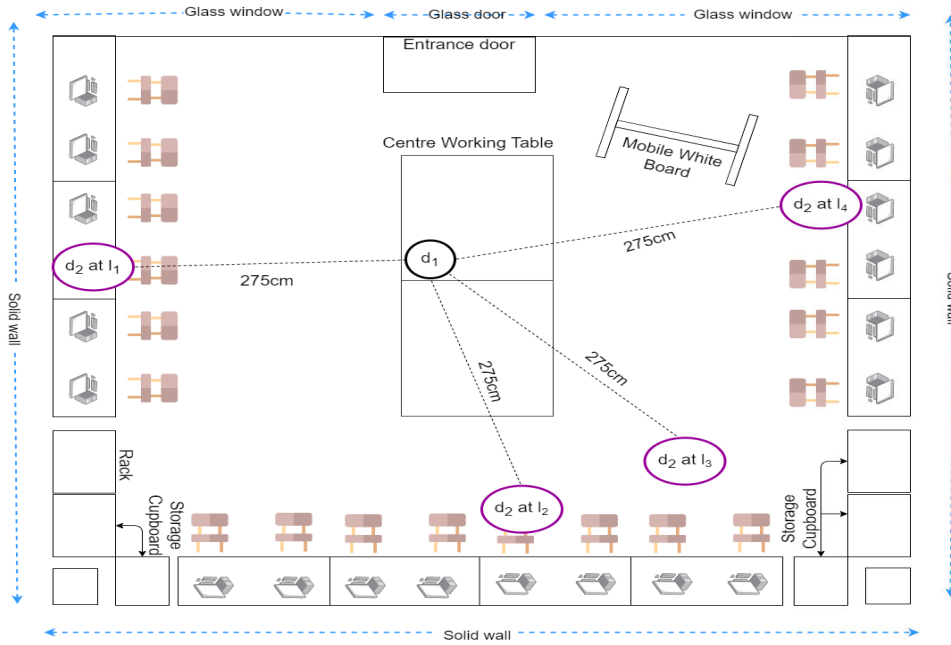
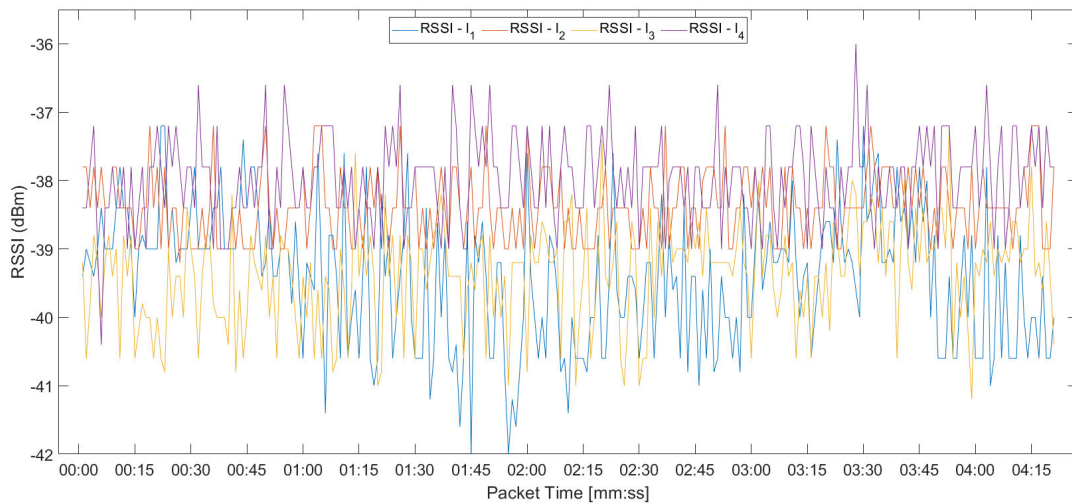**FIGURE 8.** Experimental lab layout (fixed distance and varying location scenarios).

FIGURE 9. RSSI for four different locations (l_1, l_2, l_3, l_4) - Fixed distance between transmitter and receiver.

arithmetic mean of all five readings at each timestamp to observe the fluctuations in RSSI having stationary devices till 262 seconds, which can be mathematically represented as:

$$\gamma(\tau) = \frac{1}{r} \sum_{q=1}^{r} \gamma_q(\tau), \qquad (41)$$

where r represents the number of readings for each time slot, and the summation is taken over all the individual readings $\gamma_q(\tau)$ at timestamp τ, where q ranges from 1 to r. Table 5 represents the RSSI data-set of initial 20 secs collected at the first location in a laboratory environment as explained earlier in subsection V-C1. The complete data sets can be acquired on request.

Next, we tend to extend the existing experiment scenario. For this, we moved the receiver to three more locations, such as (l_1, l_2, l_3, l_4), and recorded the RSSI data having a fixed distance of 2.75 m between the transmitter and receiver device having LoS communication. Then, we processed the data using the same procedure as represented in equation 41. The behaviour of RSSI for all four locations can be seen in Figure 9.

VI. RESULTS AND DISCUSSION

In this work, conventional estimation techniques have been implemented to estimate the LP-IoT wireless channel. For this, we investigated the implementation of LSE and MLE
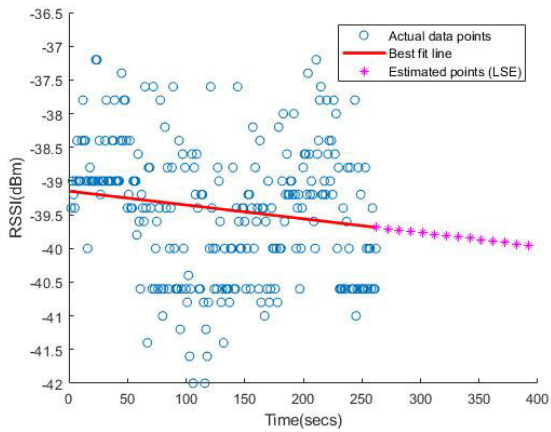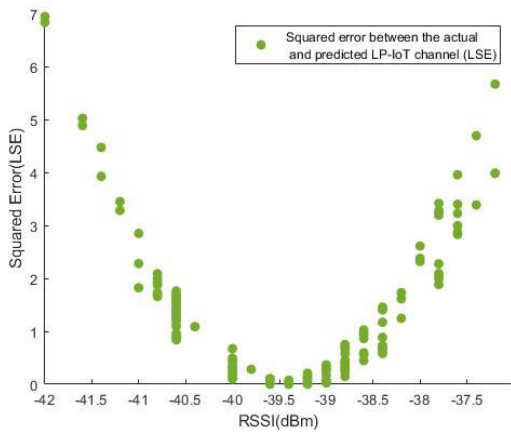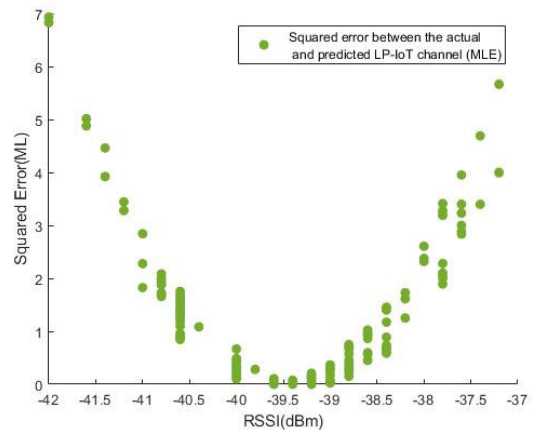
(a) LSE for LP-IoT wireless channel.
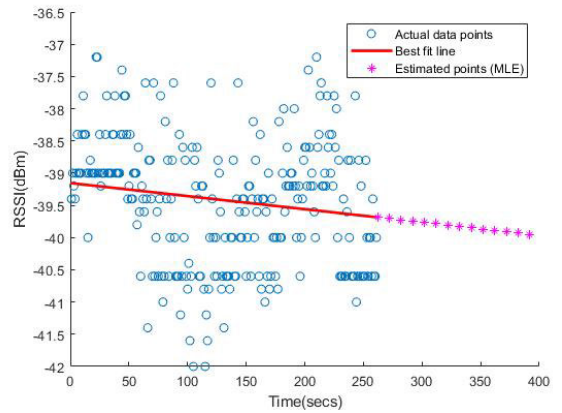
(b) MLE for LP-IoT wireless channel.

FIGURE 10. LSE and MLE methods for LP-IoT wireless channel.
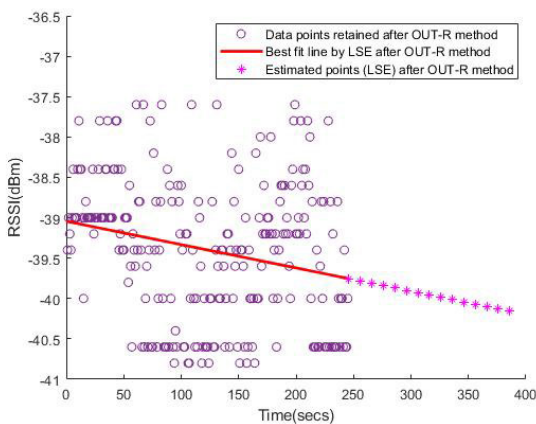

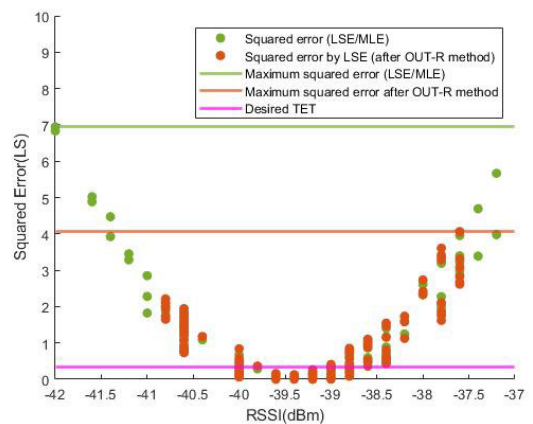(a) Squared error by LSE for LP-IoT wireless channel.

(b) Squared error by MLE for LP-IoT wireless channel.

FIGURE 11. Squared error acquired by implementing LSE and MLE for LP-IoT wireless channel.

(a) LSE for LP-IoT wireless channel after implementing OUT-R method.

(b) Squared error by LSE after implementing OUT-R method.

FIGURE 12. Least square estimation and its squared error for LP-IoT wireless channel followed by the OUT-R method.

to acquire the squared error for analysis. In other systems, the BFL acquired by LSE and MLE minimises the sum of

squares of estimation error and is the most likely function to produce the observed data. However, the squared error
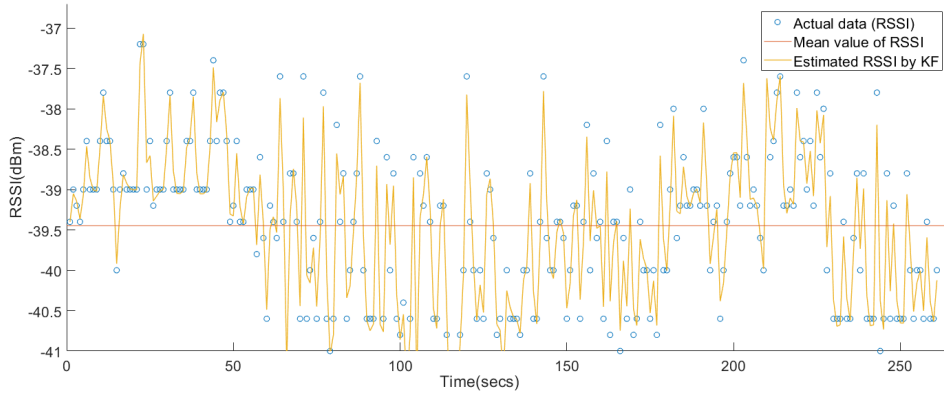
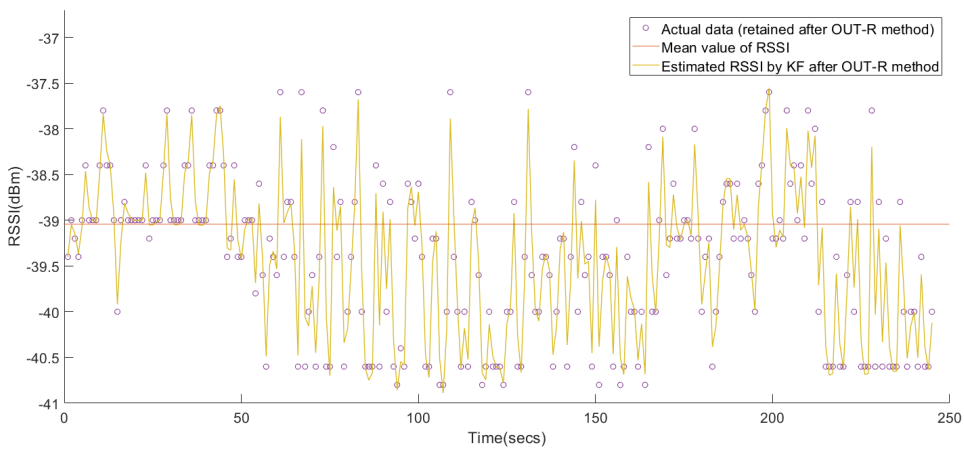**FIGURE 13.** LP-IoT channel estimation by KF using real measurements.

FIGURE 14. LP-IoT channel estimation by KF using the measurements retained after implementing OUT-R method.

(a) Squared error by KF (without OUT-R method).
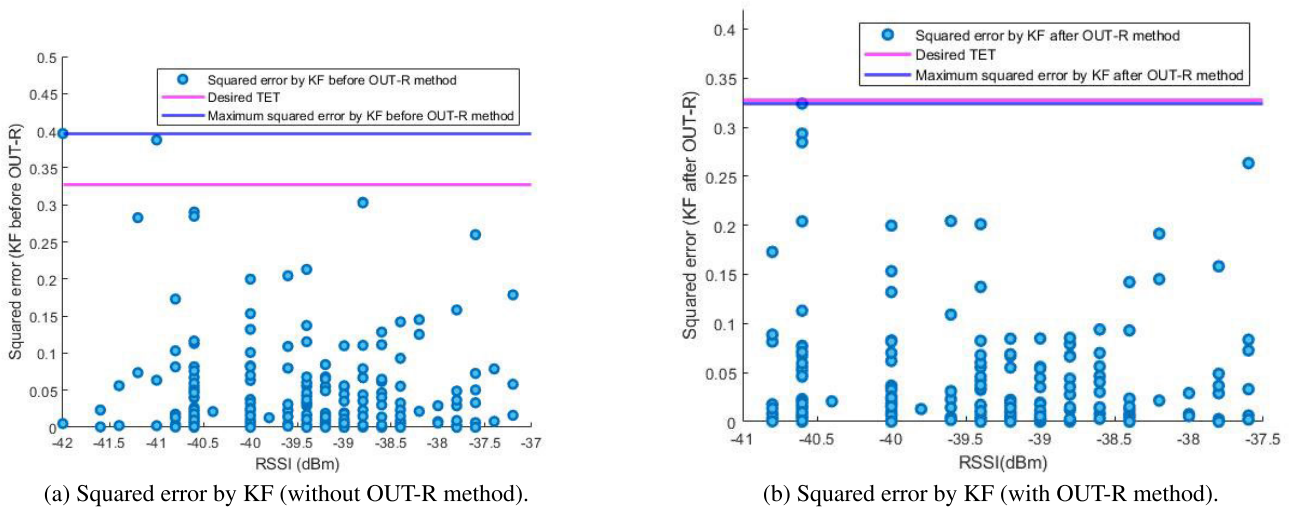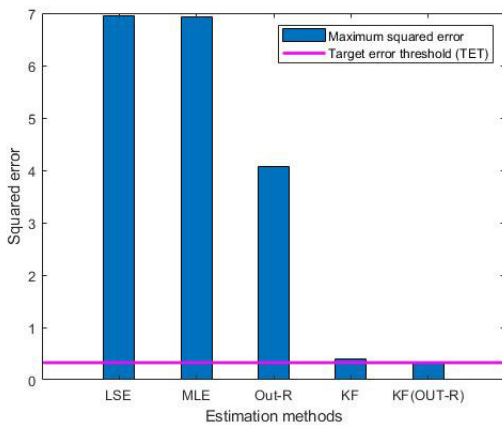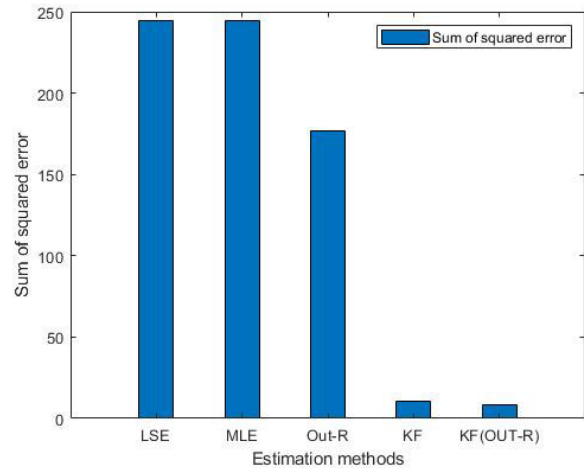

(b) Squared error by KF (with OUT-R method).

FIGURE 15. The squared error representation by the implementation of Kalman filter method with and without OUT-R method.

analysis in our implementation between the measured and estimated channel (RSSI) data indicates that the squared error must be greatly reduced to increase the estimation accuracy.

There are established procedures to minimise squared errors, but our goal is to obtain the most accurate estimation with the least processing time. Therefore, we applied LSE

(a) Squared error representation of LSE, MLE, OUT-R, KF and KF with OUT-R method.

(b) Sum of squared error representation of LSE, MLE, OUT-R, KF and KF with OUT-R method.

FIGURE 16. Visual representation of squared error and sum of squared error minimisation by implemented estimation techniques.

and MLE methods for LP-IoT channel estimation as seen in Figures 10a and 10b, which yield an estimate with a maximum squared error of approximately 6.94 *dBm* (from LSE and MLE) as shown in Figures 11a and 11b. This squared error is significantly high and may not provide an accurate estimation. An outlier removal method named OUT-R was developed and applied to reduce this squared error. This method reduced the maximum squared error of 6.94 *dBm* acquired from LSE and MLE to 4.06 *dBm*. The BFL representing the estimation after implementing the OUT-R method and its squared error representation can be seen in Figures 12a and 12b.

Next, the KF method was applied to the data retained after the OUT-R method to minimise the squared error further and achieve the TET as shown in Figure 14. The squared error acquired through implementing the KF (after the OUT-R method) gives the maximum squared error of 0.32 *dBm* approximately equal to the upper fence of pre-set error threshold (TET) of 0.32 *dBm* with a negligible difference as depicted in Figure 15b. The illustrative representation of squared error and the sum of squared errors using the five techniques (LSE, MLE, OUT-R, KF and KF with OUT-R method) can be seen in a bar graph in Figures 16a and 16b.

Next, we investigated the direct implementation of KF using real measurements without implementing the OUT-R method. The straightforward implementation of KF without OUT-R (as shown in Figure 13) provides the maximum squared error of 0.39 *dBm* (as shown in Figure 15a) and the sum of squared error that is 10.46 *dBm*. However, the KF with OUT-R method gives the maximum squared error of 0.32 *dBm* (as shown in Figure 15b) and the sum of squared error of 8.21 *dBm*. The comparison between the implementation of KF with and without the OUT-R method indicates that the estimation accuracy of the LP-IoT wireless channel can be

TABLE 6. The representation of LP-IoT channel estimation by KF method with OUT-R, evaluated at varying receiver locations (scenario 2).

Location (Receiver)	Maximum Squared Error (dBm)	Sum of Squared Error (dBm)	Mean Squared Error (dBm)
l_2	0.12 *dBm*	3.54 *dBm*	0.01 *dBm*
l_3	0.30 *dBm*	7.96 *dBm*	0.03 *dBm*
l_4	0.2 *dBm*	6.42 *dBm*	0.02 *dBm*

improved by combining the OUT-R method with KF instead of implementing KF straightaway. Subsequently, we analysed the implementation of OUT-R at different locations, which led to the retention of all the data points (no outliers detected) according to the range of RSSI defined in the OUT-R method. Table 6 displays the outcomes of RSSI estimation conducted at different locations. The summarised estimation results for two experiment scenarios are presented in table 7. Finally, the results are compared to the findings reported in [33]. The study demonstrates that the MSE of RSSI prediction in [33] is significantly higher when compared to KF with the OUT-R method. The MSE values reported in [33] exhibit a range between 38.3*dBm* and 45.54*dBm* for all datasets. However, when employing the OUT-R approach with the KF, the MSE values range from 0.01*dBm* to 0.03*dBm*. Nevertheless, it is acknowledged that the datasets utilised in our research scenario are constrained, with intentions to expand them in future research. Furthermore, it is imperative to enhance the OUT-R technique by improving the LP-IoT RSSI ranges, considering diverse scenarios, such as fluctuating distances between LP-IoT devices and a mixed mobility environment. Furthermore, we intend to apply this channel estimate model in scenarios without direct LoS, known as Non-Line-of-Sight (NLoS) conditions. In doing so, we will include additional channel characteristics to assess and confirm the model's resilience and ability to be applied across many scenarios.
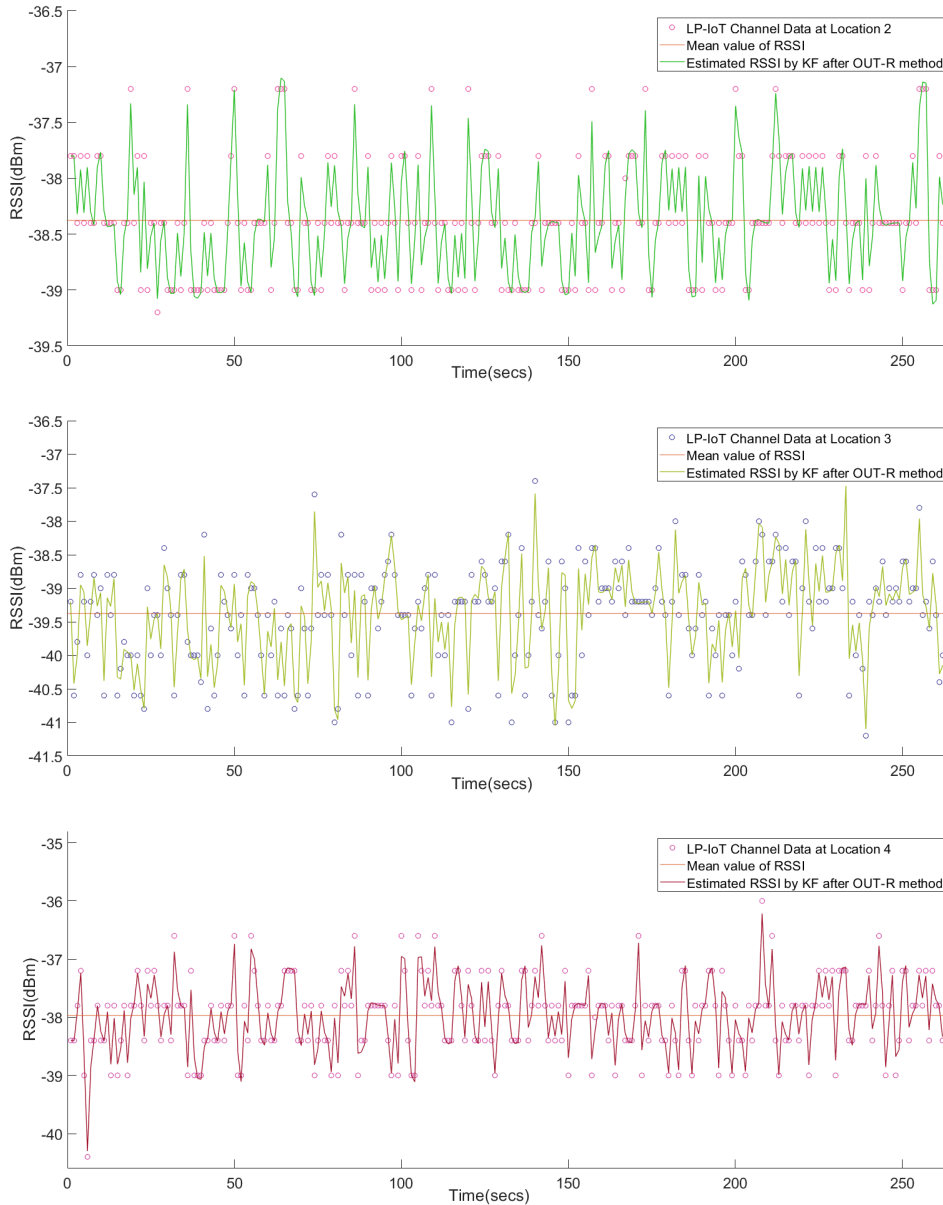
FIGURE 17. LP-IoT channel estimation by KF with OUT-R method at locations l_2, l_3, and l_4.

TABLE 7. The summerised result of two experiment scenarios for KF with OUT-R method.

Experiment Scenario	Sum of Squarred Error	Mean Squarred Error
Scenario-1 Fixed Distance	$8.21 \, dBm$	$0.03 \, dBm$
Scenario-2 Varying Location	$5.97 \, dBm$	$0.02 \, dBm$

VII. CONCLUSION AND FUTURE WORK

This study demonstrated the wireless channel estimation for low-power IoT devices based on RSSI. The LP-IoT wireless channel data was acquired by deploying two Waspmote LP-IoT devices with LoS communication in an indoor environment. The system model provided in this paper consists of a theoretical communication model and an estimation model for the LP-IoT wireless channel. The traditional estimation techniques (LSE and MLE) were utilised for the LP-IoT channel estimation to estimate the wireless channel. By implementing these estimation techniques, we analysed the squared and sum of squared errors and obtained the target error threshold (TET) by calculating the confidence interval. A novel outlier removal approach for estimating the LP-IoT channel named OUT-R was developed to meet the TET value. Using this strategy, we could reduce the estimation error but could not reach the TET value. Next, following the OUT-R method, the KF technique was applied to the retained

LP-IoT channel data to reduce further estimation error and achieve the TET. According to the KF estimation results, the squared error was 0.32 *dBm*, closer to the TET squared error of 0.32 *dBm*. The conclusion of this study shows that the combination of the OUT-R technique and KF can produce reliable estimations for LP-IoT wireless channels.

While the OUT-R method with KF demonstrates favorable results across different experiment scenarios, its performance in a dynamic indoor environment where LP-IoT devices relocate arbitrarily has yet to be examined. This investigation is necessary to enhance the model's versatility, generalisability, and robustness. Our future research plans also include exploring specialised algorithms tailored to IoT environments, integrating additional channel parameters, and assessing their applicability across diverse communication devices. It is important to note that the LP-IoT wireless channel data utilised in this study is derived from a limited dataset. Nevertheless, conducting evaluations on more extensive datasets is imperative to investigate the effectiveness of the employed technique.

An interesting aspect of this work is to integrate Machine Learning (ML) with the extensive datasets in LP-IoT wireless channel estimation, particularly in various indoor experimental scenarios. This aspect is also planned for future studies.

REFERENCES

[1] A. Rayes and S. Salam, "Internet of Things (IoT) overview," in *Internet of Things From Hype To Reality*, Springer, 2019, pp. 1–35.

[2] A. Khanna and S. Kaur, "Internet of Things (IoT), applications and challenges: A comprehensive review," *Wireless Pers. Commun.*, vol. 114, no. 2, pp. 1687–1762, Sep. 2020.

[3] (2023). *Increment in IoT Connections*. Accessed: Jan. 25, 2023. [Online]. Available: https://www.statista.com/statistics/1183457/iot-connected-devices-worldwide/

[4] S. Bansal and D. Kumar, "IoT ecosystem: A survey on devices, gateways, operating systems, middleware and communication," *Int. J. Wireless Inf. Netw.*, vol. 27, no. 3, pp. 340–364, Sep. 2020.

[5] S. Al-Sarawi, M. Anbar, K. Alieyan, and M. Alzubaidi, "Internet of Things (IoT) communication protocols: Review," in *Proc. 8th Int. Conf. Inf. Technol. (ICIT)*, May 2017, pp. 685–690.

[6] N. Baccour, A. Koubâa, H. Youssef, and M. Alves, "Reliable link quality estimation in low-power wireless networks and its impact on tree-routing," *Ad Hoc Netw.*, vol. 27, pp. 1–25, Apr. 2015.

[7] H. A. H. Alobaidy, M. Jit Singh, M. Behjati, R. Nordin, and N. F. Abdullah, "Wireless transmissions, propagation and channel modelling for IoT technologies: Applications and challenges," *IEEE Access*, vol. 10, pp. 24095–24131, 2022.

[8] H.-J. Audéoud and M. Heusse, "Quick and efficient link quality estimation in wireless sensors networks," in *Proc. 14th Annu. Conf. Wireless On-Demand Netw. Syst. Services (WONS)*, Feb. 2018, pp. 87–90.

[9] D. Dinev and A. Haka, "RSSI study of wireless Internet of Things technologies," *J. Phys., Conf. Ser.*, vol. 2339, no. 1, Sep. 2022, Art. no. 012014.

[10] S. Dolha, P. Negirla, F. Alexa, and I. Silea, "Considerations about the signal level measurement in wireless sensor networks for node position estimation," *Sensors*, vol. 19, no. 19, p. 4179, Sep. 2019.

[11] F. Subhan, A. Khan, S. Saleem, S. Ahmed, M. Imran, Z. Asghar, and J. I. Bangash, "Experimental analysis of received signals strength in Bluetooth low energy (BLE) and its effect on distance and position estimation," *Trans. Emerg. Telecommun. Technol.*, vol. 33, no. 2, e3793, Feb. 2022.

[12] H. C. So and L. Lin, "Linear least squares approach for accurate received signal strength based source localization," *IEEE Trans. Signal Process.*, vol. 59, no. 8, pp. 4035–4040, Aug. 2011.

[13] V. Kumar Singh, M. F. Flanagan, and B. Cardiff, "Maximum likelihood channel path detection and MMSE channel estimation in OTFS systems," in *Proc. IEEE 92nd Veh. Technol. Conf. (VTC-Fall)*, Nov. 2020, pp. 1–5.

[14] T. Chen, B. Chen, D. Xu, Y. Ding, R. Pang, C. Gong, and H. Zhang, "Channel quality estimation with MMSE filter and Viterbi decoding for airborne communications," in *Proc. Integr. Commun., Navigat. Surveill. Conf. (ICNS) Conf.*, Apr. 2014, pp. R4-1–R4-8.

[15] F. Qin, X. Dai, and J. E. Mitchell, "Effective-SNR estimation for wireless sensor network using Kalman filter," *Ad Hoc Netw.*, vol. 11, no. 3, pp. 944–958, May 2013.

[16] S. J. Miller, "The method of least squares," Dept. Math. Dept., Brown Univ., Providence, RI, USA, Tech. Rep. RI 02912, 2006, vol. 8, pp. 1–7.

[17] I. J. Myung, "Tutorial on maximum likelihood estimation," *J. Math. Psychol.*, vol. 47, no. 1, pp. 90–100, Feb. 2003.

[18] D. Guo, S. Shamai (Shitz), and S. Verdu, "Mutual information and minimum mean-square error in Gaussian channels," *IEEE Trans. Inf. Theory*, vol. 51, no. 4, pp. 1261–1282, Apr. 2005.

[19] P. S. Maybeck, "The Kalman filter: An introduction to concepts," in *Autonomous Robot Vehicles*. New York, NY, USA: Springer, 1990, pp. 194–204.

[20] C. M. Ramya, M. Shanmugaraj, and R. Prabakaran, "Study on ZigBee technology," in *Proc. 3rd Int. Conf. Electron. Comput. Technol.*, Apr. 2011, pp. 297–301.

[21] S. Anand, R. Muralidharan, K. Manoj, C. Shreyas, and H. Kariappa, "Rssi strength measurement in wireless sensor network with and without obstacles," in *Micro-Electronics and Telecommunication Engineering*. Singapore: Springer, 2022, pp. 593–608.

[22] K. Sangrit, J. Karnjana, S. Laitrakun, K. Fukawa, S. Fugkeaw, and S. Keerativittayanun, "Distance estimation between wireless sensor nodes using RSSI and CSI with bounded-error estimation and theory of evidence for a landslide monitoring system," in *Proc. 13th Int. Conf. Inf. Technol. Electr. Eng. (ICITEE)*, Oct. 2021, pp. 40–45.

[23] G. Onoh, S. Arinze, and I. Offia, "Development of an intelligent ZigBee technique for improving the energy efficiency and link quality of a wireless sensor network," *Eur. J. Eng. Environ. Sci.*, vol. 1, no. 1, pp. 1–8, Feb. 2022.

[24] J. Luomala and I. Hakala, "Analysis and evaluation of adaptive RSSI-based ranging in outdoor wireless sensor networks," *Ad Hoc Netw.*, vol. 87, pp. 100–112, May 2019.

[25] M. S. Ali, M. K. H. Jewel, and F. Lin, "An efficient channel estimation technique in NB-IoT systems," in *Proc. IEEE Int. Conf. Integr. Circuits, Technol. Appl. (ICTA)*, Nov. 2018, pp. 22–23.

[26] 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. Electron. Nanotechnol. (ELNANO)*, Oct. 2022, pp. 576–580.

[27] A. Balakrishnan, K. Ramana, K. Nanmaran, M. Ramachandran, V. Bhaskar, and S. Kallam, "RSSI based localization and tracking in a spatial network system using wireless sensor networks," *Wireless Pers. Commun.*, vol. 123, no. 1, pp. 879–915, Mar. 2022.

[28] J. Wen and W. Dargie, "Characterization of link quality fluctuation in mobile wireless sensor networks," *ACM Trans. Cyber-Phys. Syst.*, vol. 5, no. 3, pp. 1–24, Jul. 2021.

[29] T. Jiang, Y. Shi, J. Zhang, and K. B. Letaief, "Joint activity detection and channel estimation for IoT networks: Phase transition and computation-estimation tradeoff," 2018, *arXiv:1810.00720*.

[30] J. Ko, H. Kim, and J. Kim, "Real-time sound source localization for low-power IoT devices based on multi-stream CNN," *Sensors*, vol. 22, no. 12, p. 4650, Jun. 2022.

[31] F. Jamil, N. Iqbal, S. Ahmad, and D.-H. Kim, "Toward accurate position estimation using learning to prediction algorithm in indoor navigation," *Sensors*, vol. 20, no. 16, p. 4410, Aug. 2020.

[32] W. Kim, Y. Ahn, J. Kim, and B. Shim, "Towards deep learning-aided wireless channel estimation and channel state information feedback for 6G," *J. Commun. Netw.*, vol. 25, no. 1, pp. 61–75, Feb. 2023.

[33] N. Raj, "Indoor RSSI prediction using machine learning for wireless networks," in *Proc. Int. Conf. Commun. Syst. Netw. (COMSNETS)*, Jan. 2021, pp. 372–374.

[34] M. K. Ozdemir and H. Arslan, "Channel estimation for wireless OFDM systems," *IEEE Commun. Surveys Tuts.*, vol. 9, no. 2, pp. 18–48, 2nd Quart., 2007.

[35] R. Millar, *Maximum Likelihood Estimation and Inference: With Examples in R, SAS and ADMB*. Hoboken, NJ, USA: Wiley, Jul. 2011.

[36] K. S. Kannan and K. M. S. Arumugam, "Labeling methods for identifying outliers," *Int. J. Statist. Syst.*, vol. 10, no. 2, pp. 231–238, Oct. 2015.

[37] G. Welch and G. Bishop, *An Introduction To the Kalman Filter*, Chapel Hill, NC, USA, 2006.

[38] M. K. Simon and M. S. Alouini, *Digital Communication Over Fading Channels*, vol. 95. Hoboken, NJ, USA: Wiley, 2005.

[39] (2021). *Euler's Formula for Complex Numbers*. Accessed: Sep. 7, 2021. [Online]. Available: https://www.mathsisfun.com/algebra/eulers-formula.html

[40] S. M. Kay, *Fundamentals of Statistical Signal Processing: Estimation Theory*. Upper Saddle River, NJ, USA: Prentice-Hall, 1993.

[41] H. Aguinis, R. K. Gottfredson, and H. Joo, "Best-practice recommendations for defining, identifying, and handling outliers," *Organizational Res. Methods*, vol. 16, no. 2, pp. 270–301, Apr. 2013.

[42] M. Khodarahmi and V. Maihami, "A review on Kalman filter models," *Arch. Comput. Methods Eng.*, vol. 30, pp. 727–747, 2023.

[43] (2023). *General Characteristics of Waspmote*. Accessed: Apr. 4, 2023. [Online]. Available: https://www.libelium.com/iot-products/waspmote/

[44] (2023). *XBEE PRO Communication*. Accessed: Apr. 4, 2023. [Online]. Available: https://development.libelium.com/waspmote-technical-guide/802-15-4-zigbee-RF-modules

[45] (2021). *Waspmote-Wireless Sensor Networks Open Source Platform*. Accessed: Aug. 23, 2021. [Online]. Available: http://www.libelium.com/products/waspmote/

[46] K. Saleem, F. Y. Alfariheedi, R. Ouni, and J. Al-Muhtadi, "Cellular IoT based secure monitoring system for smart environments," in *Proc. IEEE Int. Conf. E-Health Netw., Appl. Services (HealthCom)*, Oct. 2022, pp. 1–5.

[47] Z. Ma, R. Rayhana, Z. Liu, G. G. Xiao, Y. Ruan, and J. S. Sangha, "Industrial Internet of Things (IoT) and 3D reconstruction empowered smart agriculture system," in *Proc. IEEE Int. Conf. Internet Things Intell. Syst. (IoTaIS)*, Nov. 2022, pp. 311–316.

[48] K. Saleem, A. A. Alajroosh, R. Ouni, W. Mansoor, and A. Gawanmeh, "Smart and secure IoT based remote real-time radiation detection and measurement system," in *Proc. 1st Int. Conf. Adv. Innov. Smart Cities (ICAISC)*, Jan. 2023, pp. 1–5.

[49] T. N. C. Ta, D. Pham-Khac, and Q. Le-Trung, "Development of libelium-based reconfigurable solutions for smart city applications," in *Proc. RIVF Int. Conf. Comput. Commun. Technol. (RIVF)*, Dec. 2022, pp. 566–571.

[50] N. Alturki, T. Aljrees, M. Umer, A. Ishaq, S. Alsubai, O. Saidani, S. Djuraev, and I. Ashraf, "An intelligent framework for cyber-physical satellite system and IoT-aided aerial vehicle security threat detection," *Sensors*, vol. 23, no. 16, p. 7154, Aug. 2023.

[51] P. D. P. Adi, V. Sihombing, V. M. M. Siregar, G. J. Yanris, F. A. Sianturi, W. Purba, S. P. Tamba, J. Simatupang, R. Arifuddin, Subairi, and D. A. Prasetya, "A performance evaluation of ZigBee mesh communication on the Internet of Things (IoT)," in *Proc. 3rd East Indonesia Conf. Comput. Inf. Technol. (EIConCIT)*, Apr. 2021, pp. 7–13.

[52] (2022). *XBee/XBee-PRO S1 802.15.4*. Accessed: Aug. 23, 2022. [Online]. Available: https://www.digi.com/resources/documentation/digidocs/pdfs/90000982.pdf

[53] (2021). *XCTU-Next Generation Configuration Platform for XBee/RF Solutions*. Accessed: Sep. 29, 2021. [Online]. Available: https://www.digi.com/products/embedded-systems/digi-xbee/digi-xbee-tools/xctu

[54] (2022). *XBee/XBee-PRO S1—API Mode*. Accessed: Sep. 15, 2022. [Online]. Available: https://www.digi.com/resources/documentation/Digidocs/90001456-13/concepts/c_api_mode_detailed.htm?TocPath=XBee

SAMRAH ARIF (Graduate Student Member, IEEE) received the Bachelor of Computer Science and Information Technology degree from the NED University of Engineering and Technology, Pakistan, in 2012, and the Bachelor of Computing degree (Hons.) from Charles Sturt University, Australia, in 2019, where she is currently pursuing the Ph.D. degree. She was designated to the Executive Dean's List for distinctive academic achievement at the School of Computing, Mathematics, and Engineering, Charles Sturt University. She has extensive experience in the industry, as a Software Quality Assurance Engineer. Her research interests include the Internet of Things and cyber security and machine learning. She received the Academic Excellence Award for her exceptional academic performance.

M. ARIF KHAN (Member, IEEE) received the B.Sc. degree in electrical engineering from the University of Engineering and Technology Lahore, Pakistan, the M.S. degree in electronic engineering from the GIK Institute of Engineering Sciences and Technology, Pakistan, and the Ph.D. degree in electronic engineering from Macquarie University, Sydney, Australia. He is currently a Senior Lecturer with the School of Computing, Mathematics and Engineering, Charles Sturt University, Australia. His research interests include future wireless communication technologies, smart cities, massive MIMO systems, and cyber security. He was a recipient of the Prestigious International Macquarie University Research Scholarship (iMURS) and the ICT CSIRO scholarships for the Ph.D. degree. He also has the competitive GIK Scholarship for the master's degree.

SABIH UR REHMAN (Member, IEEE) received the Bachelor of Engineering degree (Hons.) in electronics and telecommunication engineering from the University of South Australia, Adelaide, and the Ph.D. degree in wireless sensor networks from Charles Sturt University, Australia. He is currently a Senior Lecturer and the Course Director of the School of Computing, Mathematics and Engineering, Charles Sturt University. He has extensive industry experience in delivering large-scale networking solutions along with providing system administration, security, and data integration services. His research expertise is in the areas of wireless channel modeling, network planning, routing and switching, and information security. His current research interests include the Internet of Things (IoT), knowledge discovery through data mining, and distributed network security via blockchain-enabled solutions, especially in the spheres of intelligent transport systems, environmental sustainability, e-health, and precision agriculture; with the aim to positively influence social, economic and environmental sustainability of communities in rural Australia via digitally enabled solutions. He regularly publishes his research and serves as a reviewer for a number of respected journals and conferences. He is a member of the Engineers Australia (MIEAust) and the Australian Computer Society (ACS).

• • •