Scientific Research Publishing

Wireless Engineering and Technology, 2018, 9, 20-33
http://www.scirp.org/journal/wet
ISSN Online: 2152-2308
ISSN Print: 2152-2294

Localizing Jammer in an Indoor Environment by Estimating Signal Strength and Kalman Filter

Waleed Aldosari, Mohamed Zohdy

School of Engineering and Computer Science, Oakland University, Rochester Hills, USA
Email: waldosari@oakland.edu, zohdymo@oakland.edu

How to cite this paper: Aldosari, W. and Zohdy, M. (2018) Localizing Jammer in an Indoor Environment by Estimating Signal Strength and Kalman Filter. *Wireless Engineering and Technology*, **9**, 20-33.
https://doi.org/10.4236/wet.2018.92003

Received: March 6, 2018
Accepted: April 21, 2018
Published: April 26, 2018

Abstract

Localizing a jammer in an indoor environment in wireless sensor networks becomes a significant research problem due to the ease of blocking the communication between legitimate nodes. An adversary may emit radio frequency to prevent the transmission between nodes. In this paper, we propose detecting the position of the jammer indoor by using the received signal strength and Kalman filter (KF) to reduce the noise due to the multipath signal caused by obstacles in the indoor environment. We compare our work to the Linear Prediction Algorithm (LP) and Centroid Localization Algorithm (CL). We observed that the Kalman filter has better results when estimating the distance compared to other algorithms.

Keywords

Jammer Detecting, Kalman Filter (KF), Linear Prediction (LP), Centroid Localization (CL), Jammer Received Signal Strength (JRSS), Multipath Signal, Indoor Location

1. Introduction

Wireless sensor networks (WSNs) are utilized in different fields including healthcare monitoring, industrials, military, air pollution, water quality monitoring, security monitoring, wearable devices, internet of things, and more [1] [2]. WSNs are developing as multi-hop networks where each sensor gathers and transfers information to the next hop sensor until it reaches the destination node or the sink. WSNs are designed to share the communication medium, which is vulnerable to several attacks, such as a jamming attack, Danial of Service (DoS), eavesdropping, a man in the middle attack. Jamming attacks are the most severe

attacks on WSNs due to ease of launch. A jamming attack may block the sensors from communicating with their neighbor by emitting its signal with high power to prevent a legitimate node from transmitting its data [3].

Locating a jammer in WSNs is very important to support the improvement of existing countermeasures. In an indoor environment, applications using the wireless communication are rapidly increased, such as health monitoring, internet of things applications, and monitoring secured place inside the building. Because the WSNs are designed as multi-hop networks, the sensor forwards its information to the next hop node until it is received by the destination. Therefore, routing protocol is designed to find the shortest path between the sender and the sink node before the transmitter starts transmitting its collected data. By detecting the jammer location, the routing protocol is forced to avoid the jamming region which causes repeated messages due to delivery failure [4]. Other reasons for locating the jammer position are capturing, eliminating and isolating jammer from the network, or finding a stranger in a secured place.

Jamming attacks in WSNs have been intensively studied and defined as a stranger transmitting signal with high power to inject a false signal, override the legitimate node's message, or isolate nodes from the network [5]. Furthermore, the jammer may use many techniques by adjusting its frequency. Jammers are classified into two major types: frequency domain and time domain. During a frequency domain attack, the jammer transmits its radio signal towards the target by adjusting its frequency to harm the channel, or many channels, based on its jamming strategy. During time domain attack, the jammer emits its signal periodically, which means it has two states: Sleep state and jammed state. This type of jammer is more difficult to detect because when it sleeps, we cannot tell if it exists or not. Classification of jamming attacks is described as follows [6]: A constant jammer is a frequency jamming attack. In a continuous jammer attack, a jammer emits a continuous signal with random bits which makes the channel too busy for legitimate nodes to transmit their data. A random jammer transmits the constant random data to its target. This type of attack is a time domain because the jammer sends its jamming signal periodically and then switches to sleep mode. A different kind of time domain is a reactive jammer. A jammer keeps sensing the channel until it becomes active, then it starts transmitting its jamming signal. The last type of frequency jamming attack is a deceptive jammer. Unlike the constant jammer, a deceptive jammer transmits regular data towards its target.

Several algorithms have been proposed as anti-jamming attacks in wireless communication, such as the Frequency Hopping Spread Spectrum (FHSS) and the Direct Sequence Spread Spectrum (DSSS) [7]. Both FHSS and DSSS are based on a secret shared key between nodes or sensors before exchanging their information [8]. While the sensors were randomly deployed and dynamically jointed the network, the shared secret key is infeasible when the jammer is present and blocks the communication and isolated nodes from the network before agreed upon shared key. Furthermore, due to the restricted resources in

WSNs, like memory and energy, the DSSS and FHSS are not suitable.

Indoor localization is challenging due to multipath signals caused by surrounding objects. When the signal propagates between two transceivers, it may have reflected, diffracted, or scattered before being received by the next hop node. The Non-Line-of-Sight (NLOS) [9] is the signal received by the receiver node after being reflected by objects. Due to the change of the signal path and angle degree, the detecting location techniques, such as angle of arrival, time of arrival, difference time of arrival, results in wrong distance estimation. Therefore, estimating the distance using the jammer received signal strength is significantly more accurate.

Existing location detecting technology, such as a Global Position System (GPS), may not work correctly due to the weakness of the signal inside the building [10]. Some localization techniques need additional hardware such as sonar, infrared [11], Time of Arrival (ToA), Difference Time of Arrival (DToA), and Angle of Arrival (AoA) [12] [13] [14]. These are difficult due to the restricted resources in the sensor node, including energy consumption, memory, processing, and bandwidth. Most of the jammer localization algorithms are focusing on detecting jammer location in public areas or outdoor environments and according to Yu *et al.* [15] identifying jammer position using nodes located within a jammer transmission range that is being jammed by the jammer. The jammed nodes measure the distance from the jammer using received signal strength acquired from the jammer. The boundary node adjusts its transmission power more than the jammer to be able to receive jammed node messages. Unfortunately, this approach consumes node power, which is unacceptable due to the constrained resources in WSNs. Plechrinis *et al.* [16] evaluated jammer location using Packet Delivery Ratio (PDR). Nodes near to the jammer have a weaker value of PDR. This found by examining all PDR and finding the smallest PDR, which indicates a node near the jammer location. In other words, their algorithm utilized the search technique to determine the closest boundary node to the jammer. This method obtained the nearest sensor to the jammer, not the jammer location. When the jammer was using high transmission power to jam sensors, the closest boundary node was located far away from jammer's location. [17] [18] a different method involved Centroid Localization (CL) and Weighted Centroid Localization (WCL) algorithms detecting jammer position by averaging jammed node coordinates, which were located within jammer transmission range. CL and WCL are very sensitive to their location and number of isolated nodes. In this work, we proposed detecting the position of an indoor jammer by estimating jammer's received signal strength and Kalman filter. The main challenges to locating a jammer in an indoor environment are the received JRSS not being pure and have considerable noise produced by the surrounding environment. Moreover, due to signal path loss and the reflected signal, while propagating from the transmitter to the receiver, more effort of computing the jammer location was added. Therefore, the Kalman filter was utilized in this work to reduce noise, and the path loss model was used to estimate the distance between

anchor node and advisory. We compared our work to linear prediction and centroid localization to analyze the effect of noise, jammed node positions and number for locating the jammer's position.

This paper is organized as follows: the network model and types of nodes in general model and jammed launch is discussed in section 2. In section 3, the path loss model, NLOS, and Sight-of-Line (SOL) are presented. The Linear Prediction, Centroid Localization, and Kalman filter are discussed in sections 5, 6 and 7 respectively. Section 8 includes the analysis and simulation results. Section 9 concludes this paper.

2. Network Model

We considered the sensors deployed randomly over a small area in an indoor environment. All sensors in our proposal are classified as having the following characteristics:

2.1. General Network Model

Multi-hop. Each node must pass the data collected to its neighbor to the sink node.

Stationary. All nodes have fixed position and remain not change after node deployed.

Neighbor-Aware. Each node knows its neighbor position by exchanging the location information.

Location-Award. A node can detect its location coordinate after sensors are deployed.

Homogenous. The sensor has an omnidirectional antenna and transmits with the same power level.

2.2. The Effect of Jamming Signal Model

Unaffected node: All nodes that are outside the jammer's transmission range, and they can receive a packet from all their neighbors.

Jammed nodes: Any sensors within the jammer's transmission range. A node cannot receive a message from its neighbor.

Boundary node: A node can receive a packet from part of its neighbor. A boundary node can also measure the jammer Received Signal Strength from oncoming messages. We estimate the jammer position using the boundary nodes, where they can receive the jammer's received signal strength. Figure 1 shows the network model effect by the jamming signal. Nodes and jammer are distributed randomly in our network. This figure contains unjammed nodes, jammed nodes, boundary nodes, and the jammer.

3. Log-Normal Shadowing Model

Wireless communication is susceptible to several challenges as signals travel from the transmitter to the receiver. Not only can the signal suffer from noise

and interference, but also from the reflection, diffraction, and scattering [19]. Due to the multipath signal caused by obstacles and surrounding objects in an indoor place, as shown in **Figure 2**, we use a Log-distance path loss model to estimate the distance between the boundary node and the jammer. The Log-normal shadowing model is an extension to Friis free space Equation (1).

$$P_r(d) = P_t \frac{G_t G_r \lambda^2}{(4\pi d)^2} \tag{1}$$

where $P_r(d)$ is received signal power at distance d, P_t is the transmission's signal power, G_r and G_t tare the system gain, λ is the wavelength and d is the distance between sender and receiver. The Log-normal model can be presented in the following forms:

$$P_j(d)_i = P_j(d_0) - 10 * n \log \frac{d_i}{d_0} + X_\sigma \tag{2}$$

where n is the path loss exponential where there is a change from one environment to another. In an indoor place, the path loss exponential is between 2-3. X_σ is zero-mean Gaussian distributed random variable.

$$P_j(d_0) = 10 * n \log 10 d_0 \tag{3}$$

$$P_j(d)_i = P_j(d_0) - 10 * n \log \frac{d_i}{d_0} \tag{4}$$

where d_i is the estimated distance from filtered JRSS.

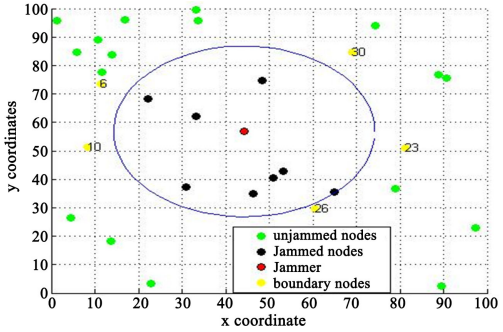

Figure 1. Network model under a jamming attack.

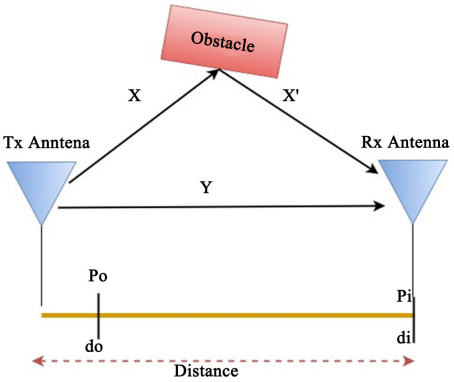

Figure 2. Log-Normal shadowing model.

4. Linear Prediction

Linear prediction is one way to predict a future value from a series of data [20] [21] [22]. In this paper, we use the linear prediction method to estimate jammer Received Signal Strength from a considerable noise caused by a multipath signal and the surrounding environment as follows.

$$\widehat{JRSS}_{LP} = \sum_{i=1}^{p} a_i * JRSS(n-i) + e[n] \tag{5}$$

$$\begin{aligned} &a_p JRSS(n-p) + a_{p-1} JRSS(n-(p-1)) + a_{p-2} JRSS(n-(p-2)) \\ &+ a_{p-3} JRSS(n-(p-3)) + a_0 JRSS(n) = JRSS(n+1) \end{aligned} \tag{6}$$

where $n = p$ is the system order. e is the error from the estimated JRSS, also called a residual signal.

$$e = \left\| JRSS - \widehat{JRSS} \right\| \tag{7}$$

Each boundary node is set to capture a series of JRSS at time N, so our data set contains N number of JRSS. By solving linear algebra for (6), we obtained the order coefficient a as follows.

$$a = \left(X^{\mathrm{T}} X\right)^{-1} X^{\mathrm{T}} Y \tag{8}$$

$$X = \begin{bmatrix} JRSS_1 & JRSS_2 & \cdots & JRSS_p \\ JRSS_2 & JRSS_3 & \cdots & JRSS_{P+1} \\ \vdots & \vdots & \ddots & \vdots \\ JRSS_{N-p-1} & JRSS_{N-p} & \cdots & JRSS_{N-1} \end{bmatrix}, a = \begin{bmatrix} a_p \\ \vdots \\ a_1 \\ a_0 \end{bmatrix}, Y = \begin{bmatrix} JRSS_{p+1} \\ JRSS_{P+2} \\ \vdots \\ JRSS_N \end{bmatrix} \tag{9}$$

where X denotes the $JRSS_{LP}$ at time instant N, and Y is the value we want to predict. **Figure 3** shows the JRSS filtered by linear prediction with three different boundary nodes, which are located in different positions. The accuracy of the estimated signal is based on how large the noise received by the boundary node, and where it is located.

5. Centroid Localization Algorithm

Centroid Localization (CL) is used to localize a sensor by averaging all nodes around the target node. The localization error using CL is based on the density and location of jammed nodes [23]. If the jammed nodes spread around the target as shown in **Figure 1**, the estimation position may be near to the original location. However, if most of the jammed nodes located on one side, the estimated position will appear on that side. CL is described as follows:

$$\left(X_j, Y_j\right) = \frac{\sum_{i=1}^{n} X_i}{N}, \frac{\sum_{i=1}^{n} Y_i}{N} \tag{10}$$

where N is the number of jammed nodes.

6. Kalman Filter

The Kalman filter was developed by Rudolf Kalman in 1960. The Kalman filter is

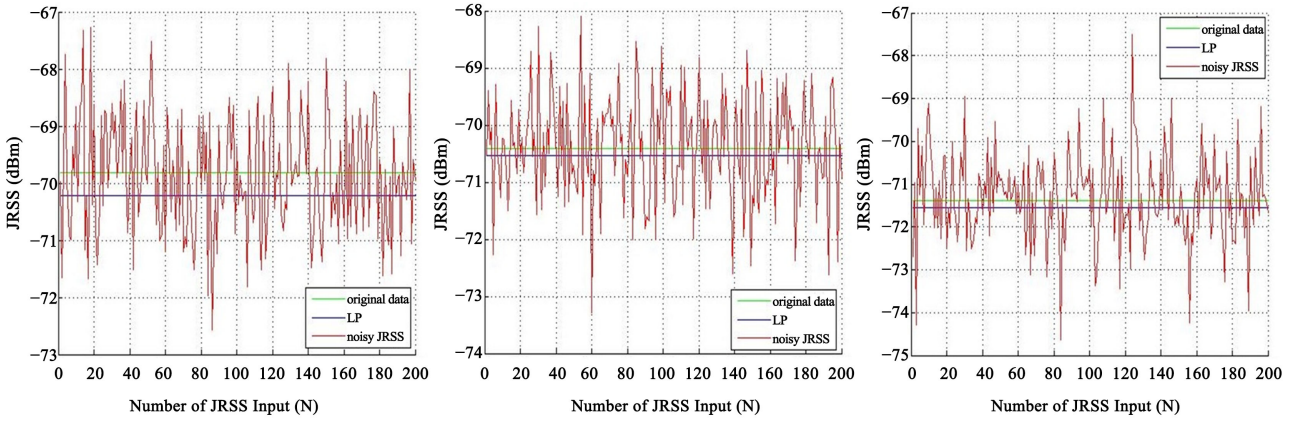

Figure 3. JRSS filtered by LP in three different boundary nodes.

a recursive estimation filter based on the linear dynamical system. It uses the past, and current estimate to predict and update current value. It has two steps to estimate the current state, prediction, and correction state [24]. We used the Kalman filter to estimate the jammer received signal strength, which has a large amount of noise caused by surrounding environment and multipath signals in an indoor place. The Kalman prediction and correction equations are as follows:

$$\hat{X}_{k|k-1} = \left(H_k Z_{k-1|k-1}\right)^{-1} \tag{11}$$

$$P_{k|k-1} = \left(H_k R_k\right)^{-1} \left(H^{\mathrm{T}}\right)^{-1} \tag{12}$$

Prediction

$$\hat{X}_{k|k-1} = A_k \hat{X}_{k-1|k-1} + B_k u_k \tag{13}$$

$$P_{k|k-1} = A_k P_{k-1|k-1} * A_k^{\mathrm{T}} + Q_k \tag{14}$$

Computing Kalman gain

$$K_k = P_{k|k-1} H_k^{\mathrm{T}} \left(H_k P_{k|k-1} H_k^{\mathrm{T}} + R_k\right)^{-1} \tag{15}$$

Updating filter

$$\hat{X}_k = \hat{X}_{k|k-1} + K_k \left(Z_k - H_k X_{k|k-1}\right) \tag{16}$$

$$P_k = P_{k|k-1} - K_k H_k P_{k|k-1} \tag{17}$$

where Z_k is the jammer's received signal strength received by a boundary node at time k, and $\hat{X}_k$ is the Kalman filter output after k times during the process. In our case, the observed JRSS at each boundary node is $\hat{X}_k$. A_k is the state transition model, H_k is the observation model, Q_k is the covariance of the process noise, R_k is the covariance of observation noise, $\boldsymbol{B}_k$ and u_k are the control input, and K is the Kalman filter gain. Because we measure the JRSS of the fixed position jammer, A_k become an identity matrix and B and u were set to zero. **Figure 4** shows the JRSS captured by three different boundary nodes and filtered by the Kalman filter. Each anchor node was located at a different distance and received the JRSS with varying amounts of noise.

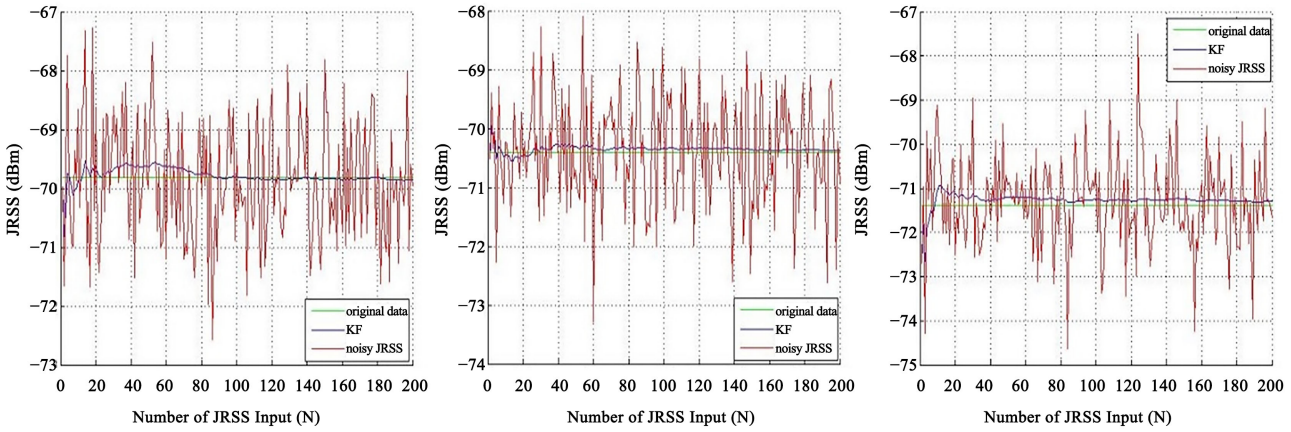

Figure 4. JRSS filtered by KF in three different boundary nodes.

7. Position Calculation

Computing jammer coordinates directly using captured JRSS resulted in the wrong position. In this section, we localize a jammer in three different methods: the Kalman filter, linear prediction, and centroid localization algorithm. To eliminate noise from the JRSS effected by surrounding environment and obstacles in an indoor place, the Kalman filter, and linear prediction come in to play. Moreover, centroid localization performs a position estimation by averaging all jammed nodes, so the noisy distance is not considered in the computation. However, CL is sensitive to the jammed nodes positions and a number of jammed nodes. In the following, we estimate the jammer distance by converting the JRSS captured by each node to distance using the Log-Normal Shadowing model (4) as follows:

$$\hat{d}_i = d_0 10^{\frac{P_j(d_0)-P_j(d)_i}{10n}} \tag{18}$$

where $\hat{d}_i$ is the estimated distance computed at each boundary node i. The Euclidean distance formula (19) is used to find the distance between the anchor and target, as shown in **Figure 5**, Where (x_i, y_i) is the known boundary node position and (x_j, y_j) is the jammer's unknown location. To locate a target position, we use the Mean Square Error (MSE). The MSE is a mathematical technique to solve linear equations with n unknown variables corresponding to n Equations (21).

$$\hat{d}_i = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2},\ i = 1, 2, 3, \cdots, n \tag{19}$$

$$\begin{cases} \hat{d}_1^2 = (x_1 - x_j)^2 + (y_1 - y_j)^2 \\ \hat{d}_1^2 = (x_1 - x_j)^2 + (y_1 - y_j)^2 \\ \quad \vdots \\ \hat{d}_n^2 = (x_n - x_j)^2 + (y_n - y_j)^2 \end{cases} \tag{20}$$

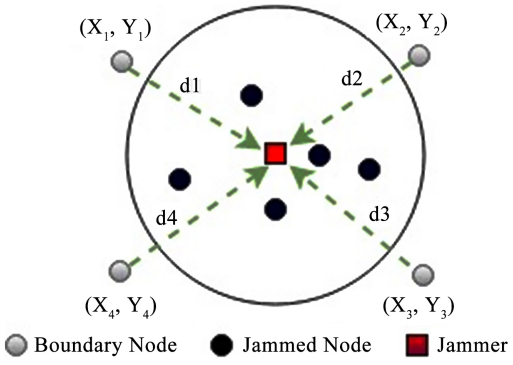

Figure 5. Illustrates distance estimation. There are four boundary nodes, five jammed nodes, and the jammer. The positions of the anchor are known.

$$\hat{x} = \left(A^{\mathrm{T}}A\right)^{-1} A^{\mathrm{T}}B \tag{21}$$

where

$$\hat{x} = \begin{bmatrix} x_j \\ y_j \end{bmatrix}, A = 2\begin{bmatrix} x_2 - x_1 & y_2 - y_1 \\ \vdots & \vdots \\ x_2 - x_1 & y_2 - y_1 \\ x_n - x_1 & y_n - y_1 \end{bmatrix}, B = \begin{bmatrix} \hat{d}_1^2 - \hat{d}_2^2 - \left(x_1^2 + y_1^2\right) + \left(x_2^2 - y_2^2\right) \\ \hat{d}_1^2 - \hat{d}_3^2 - \left(x_1^2 + y_3^2\right) + \left(x_3^2 - y_3^2\right) \\ \vdots \\ \hat{d}_1^2 - \hat{d}_n^2 - \left(x_1^2 + y_n^2\right) + \left(x_n^2 - y_n^2\right) \end{bmatrix} \tag{22}$$

where the $\left(x_i, y_i\right)$ the boundary node location and $\left(x_j, y_j\right)$ is the jammer location

8. Simulation and Results

8.1. Simulation Environment

In our network model, we simulate the effect of the jamming attack in an indoor environment to evaluate the reliability of localizing a jammer in an area of 100 m × 100 m using MATLAB. The network nodes were randomly distributed with a transmission range of 25 m and sensing a range of 15 m. The jammer location was evaluated in a different situation with a transmission range of 30m and randomly placed. We studied and evaluated three different algorithms including the Kalman filter, linear prediction, and centroid localization, and analyze the performance of each model to estimate the jammer location. To investigate the impact of JRSS samples acquired by boundary nodes, we compared KF to LP. LP is a method to predict future values and to eliminate the signal fluctuation caused by surrounding noise and multipath signals in our case. It estimates next value from a combination of past p samples, where p is system order. The main aim of LP is to compute LP coefficients to reduce the prediction error [25] [26]. The CL is utilized to evaluate the effect of jammed nodes when we estimate jammer location with different scenario compared to KF. In the experiment, we fixed the jammer location and changed the JRSS input of KF and LP. Moreover, to evaluate the robustness of CL, we placed the jammer in different locations randomly

and changed the density of the network nodes, jammer's transmission range and the nodes sensing range. For every experiment, we ran the simulation several times to evaluate the location accuracy affected by surrounding noise and a multipath signal.

8.2. Results

The mean square error (MSE) was used to evaluate the efficiency of the Kalman filter to locate the jammer in an indoor environment compared to linear prediction and centroid localization algorithms. During the experiment run time, we generated different samples of jammer received signal strength (JRSS) captured by the boundary nodes. The jammer and the nodes are randomly placed in the network. The density of the network nodes differed for each runtime to evaluate the efficiency of localizing the jammer. For the first experiment we analyzed, the network density was set to 50 nodes, the nodes and jammer were deployed randomly, and the jammer's transmission range was 35m. We studied the Kalman filter and linear prediction by increasing the number of input to 50, 100, and 200 samples as shown in Figure 6 case (a), (b), and (c) respectively. In Figure 7, KF significantly decreased the error compared to LP. For example, the mean distance error of boundary node 13 sharply reduced over the period. In case (a), KF

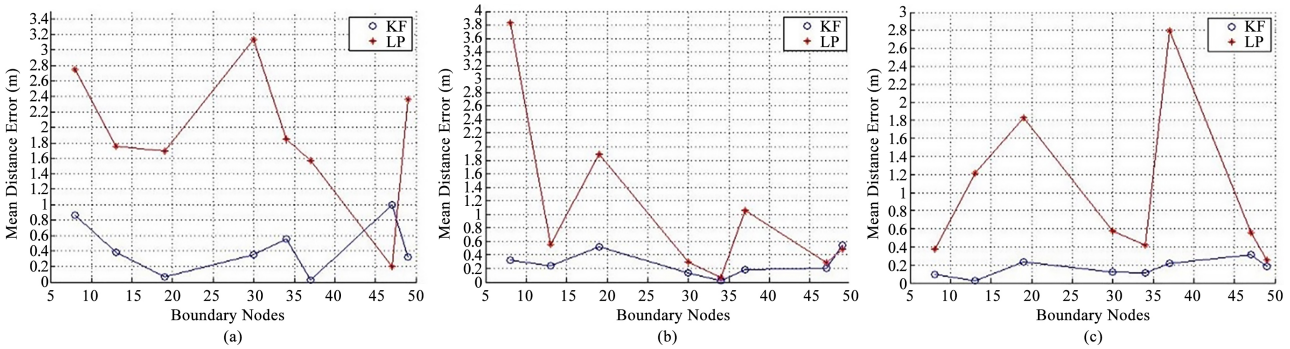

Figure 6. Mean distance error for fixed jammer position, 50 nodes, and jammer transmission range is 35 m. (a) case 1, the number of samples input (N) to KF and LP are 50; (b) case 2, N = 100; (c) case 3, N = 200.

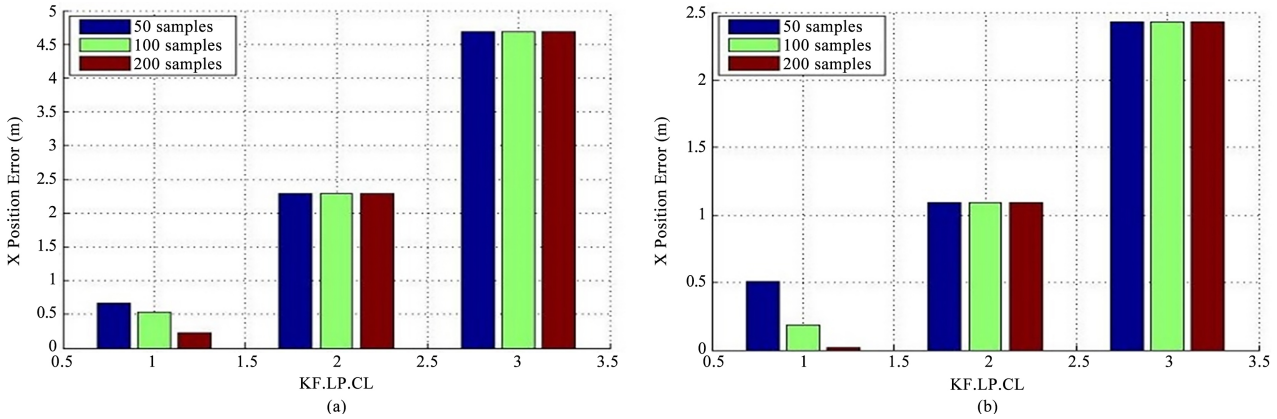

Figure 7. (x, y) Jammer coordinates error when fixed jammer location and change samples input of KF, LP, and CL to 50, 100 and 200.

computed 50 JRSSs and 200 JRSSs. The distance error between boundary node 13 and the jammer was 0.4 at 50 samples input and 0.02 at 200 JRSSs. We can see that KF has a better performance to reduce the distance error compared to LP. However, using LP and CL, the X and Y jammer coordinate error remained steady, as shown in Figure 7(a) and Figure 7(b).

Finally, we analyzed the impact of the jammer localization performance of the algorithms. Figure 8 shows the target coordinates error affected by varying the node density and jammer location. We placed the jammer in three different positions and then decreased the number of the node to 20 sensors. We can see that, because the node density does not contribute to the KF and LP output, the KF performance is better compared to LP. However, CL is based on the number of jammed nodes while the jammer location changed from one place to another. Figures 9(a)-(c) shows the influence of the number and location of jammed nodes on CL. For example, in case (a) three jammed nodes were placed inside the jammer's transmission range, and the location error is significant compared to case (b) and (c). In case (c), 8 jammed nodes contributed in CL algorithms, which gives a high accuracy of detecting the jammer location compared to LP. Table 1 compares different boundary nodes and the performance of these three algorithms: KF, LP, and CL. Table 2 shows the estimated location in three techniques and the Mean Squared Error.

Table 1. Comparison of JRSS and the estimated distance between KF, LP, and CL with 11 jammed nodes and eight boundary nodes.

Boundary node ID	Original JRSS	Noisy JRSS	KF JRSS	LP JRSS	Original Dist. (m)	Est. Dist. KF (m)	Est. Dist. LP (m)	MSE KF (m)	MSE LP (m)
4	−70.90	−70.69	−70.93	−70.53	35.09	35.21	33.64	0.08	1.02
5	−70.16	−68.93	−70.21	−70.30	32.21	32.42	32.73	0.14	0.37
16	−69.60	−68.62	−69.53	−68.96	30.20	29.99	28.07	0.15	1.50
34	−71.14	−70.57	−71.25	−71.12	36.06	36.54	35.97	0.33	0.06
38	−69.87	−68.63	−69.82	−69.67	31.16	30.98	30.45	0.12	0.50
44	−70.84	−70.86	−70.90	−70.96	34.84	35.09	35.35	0.17	0.35
45	−70.36	−69.58	−70.24	−70.46	32.99	32.52	33.37	0.33	0.26
49	−71.45	−71.38	−71.40	−72.02	37.40	37.18	39.92	0.15	1.78

Table 2. Mean (x, y) coordinates error with 11 jammed nodes and eight boundary nodes.

Algorithm	x	y	$\hat{x}$	$\hat{y}$	x-axis MSE (m)	y-axis MSE (m)
KF			63.61	18.64	0.19	0.11
LP	63.34	18.47	54.72	27.18	6.09	6.15
CL			67.19	19.62	2.72	0.81

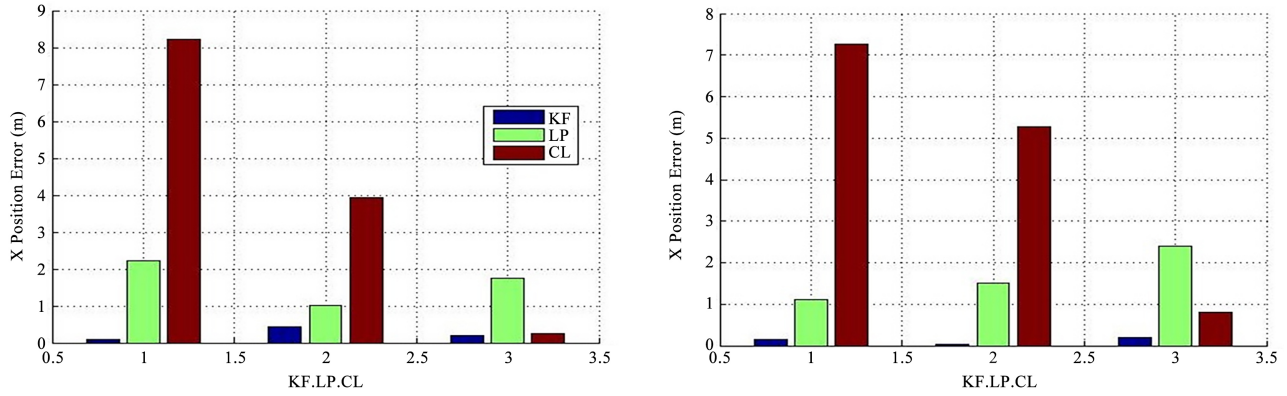

Figure 8. Illustrates the performance of KL, LP, and CL when jammer location randomly deployed in three different positions, and the density of node is 20 nodes.

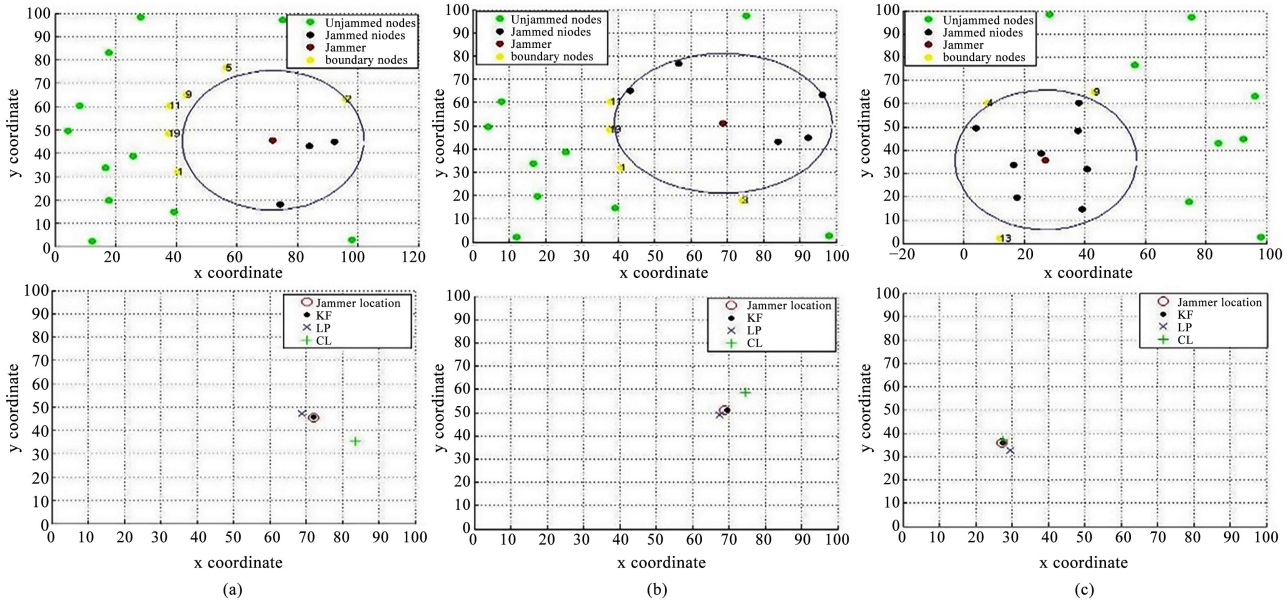

Figure 9. Estimated jammer position in three different scenarios.

9. Conclusion

In this paper, we estimate the jammer position using KF, and we compared its performance with similar algorithms, such as LP and CL. The mean distance error is very small in KF compared to LP. The CL shows better performance than LP when the jammed nodes distributed around the jammer. LP remained steady over the changes in the samples of JRSS, the density of the networks, and the location of the jammer. The KF performed better when the vast samples were taken to KF as an input and can detect the target with high accuracy compared to LP and CL.

References

[1] Kharrufa, H., Al-Kashoash, H. and Kemp, A.H. (2018) A Game Theoretic Optimization of RPL for Mobile Internet of Things Applications. *IEEE SENSORS Journal*, **18**, 2520-2530. https://doi.org/10.1109/JSEN.2018.2794762

[2] Yang, X., Chen, P.P., Gao, S.W. and Niu, Q. (2018) CSI-Based Low-Duty-Cycle Wireless Multimedia Sensor Network for Security Monitoring. *Electronics Letters*, *IET Journals & Magazines*, **54**, 323-324. https://doi.org/10.1049/el.2017.2515

[3] Zhu, J., Zou, Y.L. and Zheng, B.Y. (2017) Physical-Layer Security and Reliability Challenges for Industrial Wireless Sensor Networks. *IEEE Access*, *IEEE Journals & Magazines*, **5**, 5313-5320.

[4] Wei, X.L., Wang, Q.P., Wang, T.X. and Fan, J.H. (2017) Jammer Localization in Multi-Hop Wireless Network: A Comprehensive Survey. *IEEE Communications Surveys & Tutorials*, **19**, 765-799.

[5] Das, R., Bal, S., Das, S., Sarkar, M.K., Majumder, D., Chakraborty, A. and Majumder, K. (2016) Performance Analysis of Various Attacks under AODV in WSN & MANET Using OPNET 14.5. *IEEE 7th Annual Ubiquitous Computing*, *Electronics & Mobile Communication Conference* (*UEMCON*), *IEEE Conferences*, 1-9.

[6] Mpitziopoulos, A., Gavalas, D., Konstantopoulos, C. and Pantziou, G. (2009) A Survey on Jamming Attacks and Countermeasures in WSNs. *IEEE Communications Surveys & Tutorials*, **11**, 42-56. https://doi.org/10.1109/SURV.2009.090404

[7] Alagil, A., Alotaibi, M. and Liu, Y. (2016) Randomized Positioning DSSS for Anti-Jamming Wireless Communications. 2016 *International Conference on Computing*, *Networking and Communications* (*ICNC*), *IEEE Conferences*, 1-6.

[8] Strasser, M., Pöpper, C., Capkun, S. and Cagalj, M. (2008) Jamming-Resistant Key Establishment Using Uncoordinated Frequency Hopping. *IEEE Symposium on Security and Privacy* (sp 2008), *IEEE Conferences*, 64-78.

[9] Khan, M.M., Abbasi, Q.H., Alomainy, A. and Hao, Y. (2011) Study of Line of Sight (LOS) and None Line of Sight (NLOS) Ultra Wideband Off-Body Radio Propagation for Body Centric Wireless Communications in Indoor. *Proceedings of the* 5*th European Conference on Antennas and Propagation* (*EUCAP*), 110-114.

[10] Bulusu, N., Heidemann, J. and Estrin, D. (2000) GPS-Less Low Cost Outdoor Localization for very Small Devices. *IEEE Personal Communications*, *IEEE Journals & Magazines*, **7**, 28-34. https://doi.org/10.1109/98.878533

[11] Dakhlallah, T.K., Zohdy, M.A. and Salim, O.M. (2011) Application of Sensor Similarity, Complementarity and Type-2 Fuzzy Logic to a Dynamic Security Monitoring System. *IEEE Conferences*, *Proceedings of the* 2011 *IEEE National Aerospace and Electronics Conference* (*NAECON*), 25-31. https://doi.org/10.1109/NAECON.2011.6183072

[12] Tian, H., Wang, S. and Xie, H.Y. (2007) Localization Using Cooperative AOA Approach. *IEEE Conferences*, *International Conference on Wireless Communications*, *Networking and Mobile Computing*, 2416-2419.

[13] Rong, P. and Sichitiu, M.L. (2006) Angle of Arrival Localization for Wireless Sensor Networks. *IEEE Conferences*, 3*rd Annual IEEE Communications Society on Sensor and Ad Hoc Communications and Networks*, **1**, 347-382.

[14] Shen, G.W., Zetik, R., Yan, H.H., Hirsch, O. and Thomä, R.S. (2010) Time of Arrival Estimation for Range-Based Localization in UWB Sensor Networks. *IEEE International Conference on Ultra-Wideband*, **2**, 1-4. https://doi.org/10.1109/ICUWB.2010.5614041

[15] Kim, Y.S., Mokaya, F., Chen, E. and Tague, P. (2012) All Your Jammers Belong to us—Localization of Wireless Sensors under Jamming Attack. *IEEE International Conference on Communications* (*ICC*), *IEEE Conferences*, 949-954.

[16] Pelechrinis, K., Koutsopoulos, I., Broustis, I. and Krishnamurthy, S.V. (2009) Lightweight Jammer Localization in Wireless Networks: System Design and Imple-

mentation. *GLOBECOM* 2009-2009 *IEEE Global Telecommunications Conference*, 1-6.

[17] Xie, S.G., Hu, Y.J. and Wang, Y. (2014) Weighted Centroid Localization algorithm based on least square for wireless sensor networks. *IEEE Conferences.* 2014 *IEEE International Conference on Consumer Electronics*, China, 1-4.

[18] Blumenthal, J., Grossmann, R., Golatowski, F. and Timmermann, D. (2007) Weighted Centroid Localization in Zigbee-Based Sensor Networks. 2007 *IEEE International Symposium on Intelligent Signal Processing, Alcala de Henares*, 28 November-1 December 2007, 1-6. https://doi.org/10.1109/WISP.2007.4447528

[19] Goldsmith, A. (2005) Wireless Communications. Stanford University, Cambridge University Press, 24-46.

[20] Chetupalli, S.R. and Sreenivas Thippur, V. (2015) Successive Approximation Algorithm for LPC Estimation Using Sparse Residual Constraint. *Twenty First National Conference on Communications* (*NCC*), 1-5.

[21] Pasha, S., Ritz, C. and Zou, Y.X. (2017) Spatial Multi-Channel Linear Prediction for Dereverberation of Ad-Hoc Microphones. *Asia-Pacific Signal and Information Processing Association Annual Summit and Conference*, Kuala Lumpur, 12-15 December 2017, 1696-1700.

[22] Makhoul, J. (1975) Linear Prediction: A Tutorial Review. *Proceedings of the IEEE*, **63**, 561-580. https://doi.org/10.1109/PROC.1975.9792

[23] Liu, H.B., Xu, W.Y., Chen, Y.Y. and Liu, Z.H. (2009) Localizing Jammers in Wireless Networks. 2009 *IEEE International Conference on Pervasive Computing and Communications*, *IEEE Conferences*, 1-6.

[24] Huo, L. and Wang, Z. (2017) A Target Tracking Algorithm using Grey Model Predicting Kalman Filter in Wireless Sensor Networks. 2017 *IEEE International Conference on Internet of Things* (*iThings*) *and IEEE Green Computing and Communications* (*GreenCom*) *and IEEE Cyber*, *Physical and Social Computing* (*CPSCom*) *and IEEE Smart Data* (*SmartData*), Exeter, 21-23 June 2017, 604-610.

[25] Braun, S. and Habets, E.A.P. (2018) Linear Prediction Based Online Dereverberation and Noise Reduction using Alternating Kalman Filters. *IEEE/ACM Transactions on Audio*, *Speech*, *and Language Processing*, 1.

[26] Zhang, H., Zhang, X. and Sung, D.K. (2015) Lightweight Self-Adapting Linear Prediction Algorithms for Wireless Sensor Networks. *IEEE Sensors Journal*, **15**, 3050-3058.