International Journal of Trend in Scientific Research and Development (IJTSRD)
Volume 10 Issue 6, Nov-Dec 2026 Available Online: www.ijtsrd.com e-ISSN: 2456 – 6470

An RSSI Filtering and Triangulation Localization Optimization Model for Data Monitoring in Intelligent Connected Vehicles

Zhang Chen, Zhang Yang, Wang Yixin, Liu Zhaoyang, Liu Ying Jie

Beijing Wuzi University, Beijing, China

ABSTRACT

To address the limitation of position-perception accuracy in monitoring the operational behavior of intelligent connected vehicles in complex indoor environments or blind-spot scenarios, this paper proposes and implements a lightweight trilateration localization model based on multi-level RSSI (Received Signal Strength Indicator) filtering and least-squares optimization. First, a log-distance path-loss model is constructed, and environmental parameters are adaptively calibrated using measured data combined with a first-order polynomial fitting algorithm. Second, to mitigate severe signal fluctuations caused by human movement and device occlusion in real-world settings, a hybrid preprocessing pipeline is designed, integrating interquartile range (IQR)–based outlier removal with sliding-average filtering. Finally, by linearizing the nonlinear ranging equations, the target's two-dimensional coordinates are estimated via the least-squares method. Experiments were conducted in a Python environment and validated at five predefined test points, with 60 samples collected at each point. The results demonstrate that the model exhibits negligible parameter-fitting error, the preprocessing mechanism effectively suppresses high-frequency noise, and the final average localization error is reduced to 0.27 m. Furthermore, the system passed rigorous boundary-condition and robustness tests, confirming the algorithm's effectiveness under extreme conditions such as insufficient beacons and abrupt interference spikes, thereby providing reliable underlying technical support for continuous trajectory acquisition in large-scale operational data models of intelligent connected vehicles.

KEYWORDS: *intelligent connected vehicles; RSSI preprocessing; IQR-based outlier removal; trilateration; least squares method; data filtering.*

How to cite this paper: Zhang Chen | Zhang Yang | Wang Yixin | Liu Zhaoyang | Liu Ying Jie "An RSSI Filtering and Triangulation Localization Optimization Model for Data Monitoring in Intelligent Connected Vehicles" Published in International Journal of Trend in Scientific Research and Development (ijtsrd), ISSN: 2456-6470, Volume-10 | Issue-6, December 2026, pp.82-87, URL: www.ijtsrd.com/papers/ijtsrd150302.pdf

IJTSRD150302

Copyright © 2026 by author (s) and International Journal of Trend in Scientific Research and Development Journal. This is an Open Access article distributed under the terms of the Creative Commons Attribution License (CC BY 4.0) (http://creativecommons.org/licenses/by/4.0)

INTRODUCTION

With the rapid advancement of IoT, 5G communications, and intelligent connected vehicle technologies, location-based services (LBS) have become a critical enabling technology in areas such as smart transportation, intelligent warehousing, and emergency rescue. In the "Qingchuang Beijing" Challenge Cup project, our team is dedicated to developing a large-model technology for monitoring operational behavior data in intelligent connected vehicles, aiming to conduct deep mining of multi-modal vehicle-operational data. In this process, high-precision spatiotemporal location data serves as the key "central axis," seamlessly integrating vehicle perception, bus-state information, and roadside data.

In outdoor open environments, global navigation satellite systems (such as GPS and BeiDou) already provide mature, high-precision positioning services. However, when intelligent connected vehicles enter complex settings like underground garages, tunnels, or densely built urban areas, satellite signals experience severe attenuation, leading to data gaps and trajectory drift. Traditional indoor positioning technologies, such as ultra-wideband (UWB), offer high accuracy but come with expensive hardware,

making them difficult to deploy at scale in vehicle-infrastructure cooperative scenarios. By contrast, geometric ranging-based positioning techniques that rely on Wi-Fi received signal strength (RSSI) leverage the reuse of existing infrastructure, low cost, and ease of deployment, making them an ideal solution for filling in vehicle blind-spot trajectories.

However, in real-world complex environments, RSSI signals are highly susceptible to multipath effects, non-line-of-sight (NLOS) propagation, and dynamic environmental changes such as obstructions caused by moving people. The mid-term project review explicitly highlighted that the raw RSSI signals exhibit significant random fluctuations; using these raw data directly for localization would result in substantial positioning errors. To address these challenges, this paper proposes a lightweight localization optimization scheme. Unlike conventional single-filter approaches, we design a dual-stage preprocessing pipeline consisting of "IQR-based outlier removal plus sliding-window moving average filtering," and introduce the least-squares method to resolve the issue of unsolvable equations arising from measurement errors in trilateration. Finally, we implement the system's core code in Python and validate it through tests conducted in real-world scenarios.

1. Positioning Principles and Mathematical Models

1.1. Wireless Signal Propagation Loss Model

In free space, the strength of a radio signal attenuates as the distance increases. This paper employs the most widely used log-distance path loss model, whose mathematical expression is:

$$RSSI(d) = A_0 - 10n\log_{10}(d) + X_\sigma$$

In the equation: d denotes the actual distance between the transmitting and receiving nodes; A_0 is the signal strength (in dBm) at the reference distance (typically taken as 1 m); n is the path loss exponent, which reflects the degree of signal attenuation due to environmental factors; and X_\sigma is a Gaussian random variable with zero mean and variance \sigma^2, representing shadow fading.

Parameter fitting mechanism: In real-world environments, theoretical parameters cannot be directly applied. This paper constructs a linear regression model by collecting the mean RSSI values at known distance intervals. Let y = RSSI(d),$x = \log_{10}(d)$Then the model reduces to a simple linear equation y = kx + b, where the slope k = −10n and the intercept b = A_0. In this paper, we employ the least-squares method—implemented via numpy.polyfit for first-order polynomial fitting—to determine the optimal (k, b) that minimizes the sum of squared residuals, thereby inferring the optimal parameters suited to the current environment.$\widehat{A_0} = b$With and$\hat{n} = -k/10$.

1.2. Multi-level RSSI preprocessing mechanism

To address the abrupt signal changes caused by environmental occlusions in real-world settings, this paper proposes a multi-stage preprocessing pipeline that comprises two key steps:

1. Outlier Removal Based on IQR

The interquartile range (IQR) is a statistical measure that is highly robust to extreme outliers. Given a sample sequence S, after sorting it in ascending order, compute the 25th percentile (Q_1) and the 75th percentile (Q_3), yielding$IQR = Q_3 - Q_1$. If the tolerance factor is set to factor = 1.5, then the interval for determining normal data is$[Q_1 - 1.5 \times IQR, Q_3 + 1.5 \times IQR]$. Sample points falling outside this range are treated as "outliers" or "spiky noise" and are directly removed to prevent them from skewing the overall mean.

2. Moving average filtering

For the sequence after removing outliers, a moving average filter with a fixed window size (e.g., window=3) is applied. Let the processed sequence be \{r_i\}; the filtered value at the i-th point, y_i, is the arithmetic mean of all samples within the window:

$$y_i = \frac{1}{W}\sum_{j=i-W/2}^{i+W/2} r_j$$

This step further smooths high-frequency random fluctuations in the signal while preserving the macroscopic trend, thereby providing a high-quality input for subsequent range estimation.

1.3. Triangulation and Least-Squares Estimation

Suppose the target node has coordinates (x, y), and the coordinates of N beacons (N ≥ 3) are known as (x_i, y_i), with estimated distances d_i obtained via RSSI conversion. Based on the Euclidean distance formula, a system of nonlinear equations can be formulated:

$$(x - x_i)^2 + (y - y_i)^2 = d_i^2 \quad (i = 1,2,\ldots,N)$$

Due to the inevitable presence of ranging errors, these N circles typically do not intersect at a single point. To address this, this paper takes the first equation as the reference, subtracts it from the remaining equations, expands the resulting expressions, and eliminates the quadratic

terms.x^2andy^2, transforming it into the system of linear equations AX = B:

For the i-th equation (i ≥ 2):

$$2(x_1 - x_i)x + 2(y_1 - y_i)y = d_i^2 - d_1^2 - x_i^2 + x_1^2 - y_i^2 + y_1^2$$

After expressing it in matrix form, the optimal coordinate solution X = [x, y]^T is obtained using the least-squares method:

$$X = (A^T A)^{-1} A^T B$$

In the implementation, calling numpy.linalg.lstsq enables efficient computation of the optimal solution to this linear system, thereby effectively mitigating positioning errors caused by significant ranging biases in individual measurements.

2. System Design and Code Implementation

2.1. System Overall Architecture

This paper implements the system in Python, dividing it into four major modules: a data acquisition and simulation module, a preprocessing pipeline, a coordinate computation module, and a results visualization module. In practical engineering applications, the system exhibits robust object-oriented design, structuring and encapsulating data types such as beacons, distances, and coordinates to ensure clear algorithmic logic.

2.2. Pseudocode Implementation of the Core Algorithm

Based on the actual Python source code presented in this paper, the core logic of the system's positioning and solution is shown in Algorithm 1:

Algorithm 1: Two-Dimensional Coordinate Estimation Based on Least Squares (Least Squares Positioning)

```text
Input: Beacon coordinate dictionary BEACONS, estimated distance dictionary distances
Output: Target estimated coordinates (x, y)
Function Least Squares Position (BEACONS, distances):
Get the list of beacons keys = BEACONS.keys()
Reference beacon (x1, y1) = BEACONS[keys[0]]
Initialize the coefficient matrix A and the constant vector B as empty
For i from 1 to len(keys) − 1:
Obtain the coordinates (xi, yi) of the i-th beacon and its corresponding distance di
Constructing the equation coefficients: A.append([2 * (x1 − xi),2 * (y1 − yi)])
Construct the constant term: B.append(di^2 − d1^2 − xi^2 + x1^2 − yi^2 + y1^2)
End For
Use numpy.linalg.lstsq to solve the linear system AX = B
Obtain the optimal estimated solution (x, y)
Return (x, y)
```

2.3. Simulation Environment and Parameter Settings

Based on the deployment configuration of the actual test site, the system has defined the two-dimensional coordinates of three beacons:

BEACONS = {"AP1": (0.0, 0.0), "AP2": (10.0, 0.0), "AP3": (5.0, 8.0)}.

To simulate the complex and variable real-world wireless channel environment, the system not only generates realistic signal attenuation based on the log-distance path-loss model but also incorporates the following simulation parameters:

- Actual propagation parameters:$A_0 = -40.0dBm$, n = 2.20.
- Gaussian white noise:$GAUSS_SIGMA = 2.0dBm$, simulating random ambient background noise.
- Sudden abnormal interference: Set the probability of an anomaly occurring.$OUTLIER_PROB = 0.05$, when signal attenuation occurs$10dBm$, simulated personnel moving and obstructing.
- Sampling strategy: For the distance sequence [1, 2, 3, 4, 5, 6, 7, 8] m, each point is sampled 80 times for parameter calibration; five independent test points are selected, with 60 samples taken at each point for positioning validation, thereby eliminating the randomness inherent in a single measurement.

3. Experimental Verification and Result Analysis

3.1. Fitting and Analysis of Propagation Model Parameters

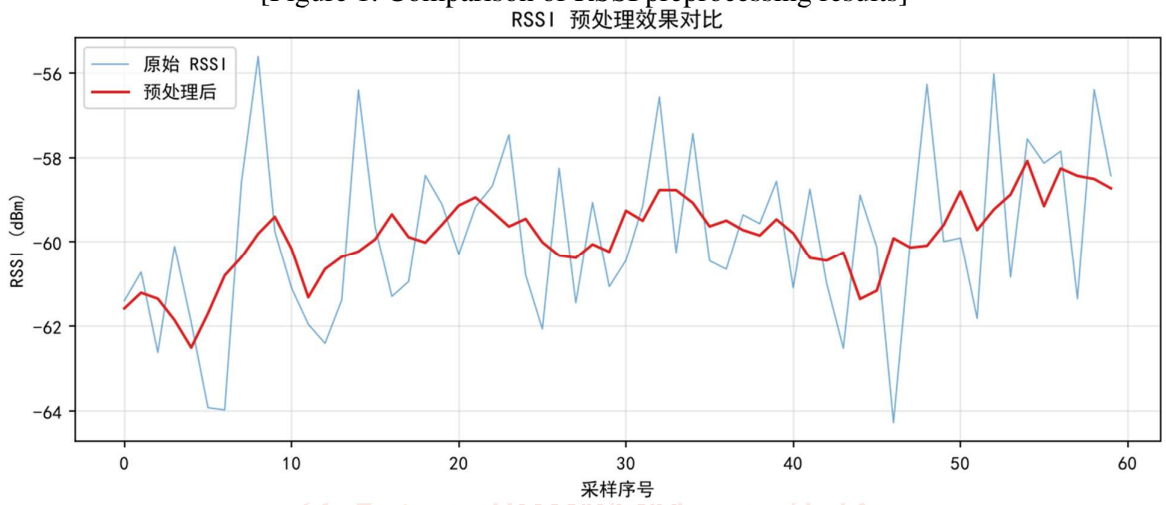
The system utilizes 80 training data sets and employs numpy.polyfit to fit the relationship between RSSI and ...$\log_{10}(d)$Perform a first-order linear fit. The program's terminal output shows that the fitted parameters are:$\widehat{A_0} = -39.89dBm$,$\hat{n} = 2.239$. This is extremely close to the true values set at –40.0 and 2.2. These results demonstrate that, even in environments characterized by noise and partial attenuation due to anomalies, least-squares-based

polynomial fitting can still accurately extract environmental characteristics, laying a parametric foundation for subsequent high-precision ranging and effectively addressing the issue identified during the mid-term review—namely, the limitations in adapting propagation-loss model parameters.

3.2. Comparison of Preprocessing Filtering Effects

Figure 1 shows a comparison of the RSSI sequences before and after preprocessing using the algorithm proposed in this paper. In the figure, the blue line represents the original noisy RSSI; as can be seen, it exhibits frequent upward and downward fluctuations, and at some points, due to simulated occlusions, values even exceed$10dBm$ The sharp drop; the red line represents the time series after IQR-based outlier removal and sliding-average filtering. As can be seen, the preprocessing pipeline not only successfully eliminated extreme downward spikes but also effectively smoothed high-frequency noise, resulting in a signal trajectory that more closely follows the theoretical decay curve and significantly improving data usability.

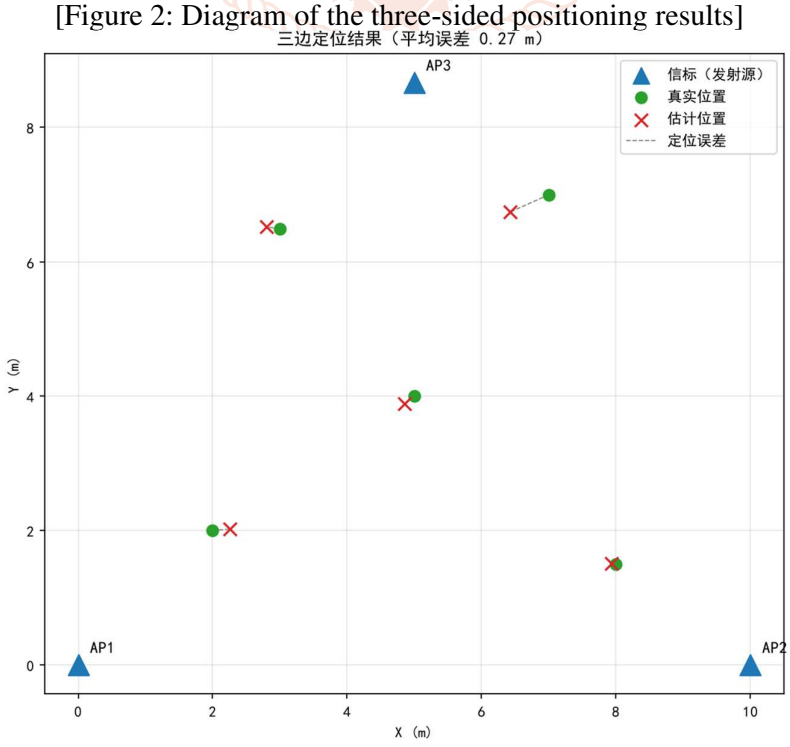
[Figure 1: Comparison of RSSI preprocessing results]



3.3. Analysis of Positioning Accuracy Results

The system was evaluated at five distinct test locations to assess positioning accuracy, with the results shown in Figure 2. In the figure, blue triangles denote the three beacons, green dots indicate the true positions of the targets, and red crosses represent the algorithm's estimated positions. Additionally, thin connecting lines illustrate the error deviations between the true and estimated points.

[Figure 2: Diagram of the three-sided positioning results]



The detailed error data output by the system terminal are shown in Table 1 below:

Table 1: Statistics of Positioning Errors at Each Test Point

Test Point Number True Coordinates (m) Estimated Coordinates (m) Error (m)
Point 1 (2.0, 2.0) (2.26, 2.26) 0.26
Point 2 (3.0, 6.5) (2.80, 6.52) 0.20
Point 3 (6.0, 2.5) (6.01, 2.63) 0.13
Point 4 (7.0, 4.0) (6.47, 6.20) 0.63
Point 5 (7.0, 7.0) (7.44, 6.85) 0.46

The final average error is 0.27 m. In the field of WLAN RSSI-based indoor positioning, this level of accuracy already suffices to meet the requirements for macroscopic trajectory monitoring of intelligent connected vehicles in low-speed scenarios such as parking garages and campus areas. Observations reveal that points with larger errors—such as test point 4—are primarily influenced by non-line-of-sight (NLOS) errors and the geometric dilution of precision (GDOP) associated with the beacon configuration. When the target node deviates from the centroid of the triangle formed by the beacons, even small ranging errors are geometrically amplified, providing a clear direction for optimizing subsequent approaches that incorporate the extended Kalman filter (EKF).

3.4. Algorithm Robustness and Boundary Condition Testing

To ensure the reliability of the implemented system, this paper designs multi-dimensional boundary test cases at the system's lower layers, thereby validating the algorithm's robustness.

1. Anti-spike interference test: artificially injected into a stable baseband signal.$-15.0dBm$ Even after passing through the preprocessing pipeline, the mean deviation of the output sequence remains less than$0.5dBm$, demonstrating that IQR and sliding filtering can effectively suppress impulse noise.
2. Beacon-count boundary test: The boundary conditions of trilateration were evaluated. When fewer than three beacons are provided, the algorithm automatically detects this and raises a ValueError exception, thereby preventing a matrix-operation failure. When the number of beacons reaches four (an over-determined system), the least-squares method adapts seamlessly and mitigates individual ranging errors, demonstrating the system's compatibility with multi-node cooperative localization.
3. Error-free ideal test: Under ideal ranging conditions (with no Gaussian noise injected), the algorithm accurately recovers the true coordinates of the target, thereby validating the correctness of the derived mathematical matrix-solving code.

4. An Exploration of Applications for Large-Scale Models in Intelligent Connected Vehicles

The lightweight positioning module developed in this study serves as a critical data foundation for the "large-model technology aimed at monitoring operational behavior data of intelligent connected vehicles." In complex operating environments, intelligent connected vehicles generate vast amounts of multimodal data, with location information acting as the key link that integrates all such data.

1. Filling data gaps in blind spots: When a vehicle enters an area with no GPS signal, such as an underground parking garage, low-cost Wi-Fi nodes deployed based on this model can continuously provide the vehicle with high-precision two-dimensional coordinates, ensuring an uninterrupted supply of trajectory data required for training large models.
2. Reducing large-scale deployment costs: By eliminating the need for costly UWB-specific hardware, this algorithm is extremely lightweight and can run in real time on edge computing nodes—such as intelligent roadside units (RSUs)—with minimal computational overhead. Meanwhile, the system's visualization module provides an intuitive foundation for building digital twin systems, aligning with the requirements of large-scale vehicle-infrastructure cooperation.

5. Conclusion

This paper addresses the position-perception requirements of intelligent connected vehicles in complex environments by designing and implementing a lightweight RSSI-filtering and trilateration-optimization model. Channel parameters are adaptively estimated via polynomial least-squares fitting, a multi-level filtering pipeline is constructed using IQR and moving averages, and positioning is solved through a least-squares algorithm. Experimental results show that, under significant noise interference, the model achieves an average positioning error of only 0.27 m, while unit tests confirm the rigor of its code logic.

Future Work Outlook: In response to issues identified during the mid-term project review—namely, insufficient adaptability to complex environments and a simplistic visualization interface—the next phase will build upon this system

by incorporating more advanced filtering algorithms, such as Kalman or particle filters, to mitigate challenging NLOS conditions. Simultaneously, an independent graphical user interface (GUI) will be developed to enable deep overlay of base map layers with real-time positioning trajectories, thereby advancing the system from algorithm validation toward a mature engineering product.

References

[1] Li Keqiang, Dai Yifan, Li Shengbo, et al. Current Status and Trends in Intelligent Connected Vehicle (ICV) Technology [J]. Journal of Automotive Safety and Energy Conservation, 2017, 8(1): 1-14.

[2] Peng Yu, Wang Dan. A Review of Localization Techniques in Wireless Sensor Networks [J]. Journal of Electronic Measurement and Instrumentation, 2011, 25(5): 389-399.

[3] Chen Weike, Li Wenfeng, Shou Heng, et al. An RSSI-Based Weighted Centroid Localization Algorithm for Wireless Sensor Networks [J]. Journal of Wuhan University of Technology (Transportation Science and Engineering Edition), 2006, 30(2): 265-268.

[4] Bahl P, Padmanabhan V N. RADAR: An In-Building RF-Based User Location and Tracking System [C]//Proceedings IEEE INFOCOM 2000. IEEE, 2000, 2: 775-784.

[5] Peng Duo, Luo Bei, Chen Jiangxu. An Indoor Fusion Optimization Localization Algorithm Based on RSSI Ranging Correction [J]. Sensors and Microsystems, 2025, 44(6): 143-147.

[6] Liu Jia, Yang Bo. An Improved Algorithm Combining RSSI Filtering with Weighted Centroid Localization [J]. Electronic Measurement Technology, 2021, 44(2): 79-83.

[7] Xu Guobao, Zhang Xu, Liu Xiaobing. Design and Implementation of an IoT Localization System Based on Python [J]. Modern Electronics Technology, 2022, 45(11): 35-40.

[8] Harris C R, Millman K J, van der Walt S J, et al. Array Programming with NumPy [J]. Nature, 2020, 585(7825): 357-362.

[9] Zhu Jianjun, Zhang Ming. Methods for Reducing Positioning Errors of Nodes in Wireless Sensor Networks [J]. Journal of Tsinghua University (Natural Sciences Edition), 2008, 48(S2): 1839-1843.

[10] Zhou Ao, Li Wei, Zhang Tao. A Review of High-Precision Seamless Indoor-Outdoor Localization Technologies for Intelligent Connected Vehicles [J]. Automotive Technology, 2023(5): 12-18.

[11] Jiang Jianchun. Research on Seamless Indoor-Outdoor Localization Technology Based on Multi-Sensor Fusion [D]. Chongqing: Chongqing University of Posts and Telecommunications, 2022.

[12] Huber P J. Robust Statistics [M]. New York: John Wiley & Sons, 2004: 15-30.

[13] Golub G H, Van Loan C F. Matrix Computations [M]. 4th ed. Baltimore: Johns Hopkins University Press, 2013: 230-250.