Investigation of Positioning Accuracy Improvement Methods for Sigfox IoT Devices through WiFi MAC Address Analysis

Masaya Watanabe[1], Takuya Nanjo[1], Tomoo Sigehuzi[2], Satoru Tada[1], Masayuki Yamauchi[1],
Takeshi Nishiyone[3], Hironori Wakaoka[3], Shuji Oji[3], Naoki Kabemoto[4], Hiroki Mori[4] ,Takeshi Tanaka[1] and Katia Vutova[5]
Hiroshima Institute of Technology, Hiroshima, Japan[1]
National Institute of Advanced Industrial Science and Technology (AIST), Tsukuba, Japan[2]
Chudenko Corporation, Hiroshima, Japan[3]
Interface Corporation, Hiroshima, Japan[4]
Institute of Electronics, Bulgarian Academy of Sciences, Sofia, Bulgaria[5]
tanaka@cc.it-hiroshima.ac.jp

Abstract: *This study presents and validates a method for achieving positioning accuracy comparable to standalone GPS (approximately 10 m) using a proprietary Sigfox WiFi positioning system that transmits only two MAC addresses per message. Due to Sigfox's severe payload limitations, conventional positioning attempts often fail or yield significant errors. Our approach introduces continuous scanning on the device to dynamically collect up to four MAC addresses, paired with meticulous backend analysis and filtering of MAC address quality using a custom Here API Validation Tool. The proposed method effectively addresses these limitations, demonstrating a 100% location acquisition rate and a median positional error of 19 meters, making it highly effective for both stationary and dynamic mobile tracking.*

***Keywords**: SIGFOX, WIFI POSITIONING, MAC ADDRESS, HERE API, LPWA, IOT, ESP32-S3*

1. Introduction

Location information is crucial across various fields, including surveying, autonomous driving, construction, and disaster response. GPS positioning, a common method for location estimation, typically incurs an error of approximately 10–30 m[1]. With the widespread adoption of IoT devices, the demand for low-cost, low-power positioning methods that utilize Sigfox, a low-power wide area network, has increased[2].

Sigfox's official Atlas WiFi service enables positioning by obtaining the MAC addresses of nearby WiFi access points. However, within the proprietary mechanism developed in this study, the number of MAC addresses that can be transmitted in a single message is strictly limited to two due to Sigfox's payload constraint of 12 bytes. Attempting to perform positioning by directly sending this limited MAC address information to the Here Positioning API often results in complete failures or significant errors (ranging from tens to hundreds of meters).

To address these challenges, this study proposes and validates a method that achieves accuracy comparable to standalone GPS positioning (approximately 10 m) using a proprietary Sigfox WiFi positioning system that transmits only two MAC addresses per message. This is achieved through meticulous scanning of the WiFi environment using NetSpot, along with quality analysis and selection of MAC addresses via a custom-built Here API Validation Tool designed for this verification.

2. GPS Positioning (Baseline accuracy verification)

A. IoT Device Configuration

We first developed an IoT device equipped with a Sigfox network module and a GPS module. The configuration combines an Arduino Uno Rev3[3] with a Sigfox shield for Arduino V2S[4] and a GPS module[5]. Detailed construction methods and the modules utilized are outlined in the literature[6].

Fig. 1 *IoT device with Sigfox network and GPS modules.*

B. GPS Positioning Results (Accuracy: Approximately 10 m)

Latitude (multiplied by 1 M) and longitude (multiplied by 1 M) data acquired from the IoT device are transmitted via email through the Sigfox cloud.

An example of the received data is given as follows:

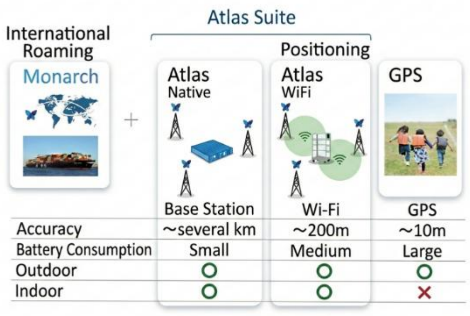
```
{
    “device” : “771E2D”,
    “time” : 1771975311,
    “seqNumber” : 954,
    “data” : “020c7a2807e640d0”,
    “slt” : 34372136,
    “slg” : 132530384
}
```

When the acquired location information was plotted on Google Maps, an error of approximately 10 m was observed between the GPS positioning (red dot) and the actual physical location (blue dot). In general, GPS positioning errors range from 10 to 30 m due to factors such as ionospheric delay and multipath effects, which align with the findings of this experiment. Therefore, this study establishes a GPS precision of approximately 10 m as the target accuracy for the Sigfox WiFi positioning system.

Fig. 2 *IoT GPS device displayed on Google Maps.*

3. Positioning using Sigfox: Mechanisms and Challenges

This section discusses the mechanisms of positioning using the Sigfox network, its accuracy, and the inherent challenges.



	Base Station	Wi-Fi	GPS
Accuracy	~several km	~200m	~10m
Battery Consumption	Small	Medium	Large
Outdoor	○	○	○
Indoor	○	○	✕

Fig. 3 Comparison of Sigfox Atlas Native and WiFi.

A. Atlas Native Positioning (Accuracy: Several Kilometers)

Atlas Native is a location measurement service that communicates with Sigfox base stations to provide geographical coordinates (latitude and longitude) and an estimated error range. This information is relayed via callbacks or the standard Sigfox API based on factors such as signal strength from transmitted messages. Although it is extremely low-power and does not require additional modules like GPS, its accuracy is limited to several kilometers, allowing only for rough area estimations. In this experiment, the indicated area spanned 90 km east-west and 110 km north-south.

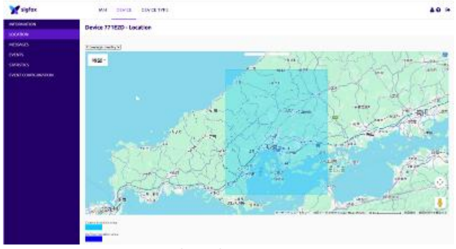


Fig. 4 *Mechanism of Sigfox Atlas location estimation.*

B. Atlas WiFi and Custom Sigfox WiFi Positioning (Here API Integration)

Sigfox's official Atlas WiFi service enables low-power acquisition of location information both indoors and outdoors by utilizing surrounding WiFi access points (BSSID/MAC addresses), provided the area has Sigfox coverage. It offers latitude, longitude, and an expected accuracy range based on MAC addresses, facilitating location identification even in environments where GPS signals are unavailable, such as indoors or in urban canyons.

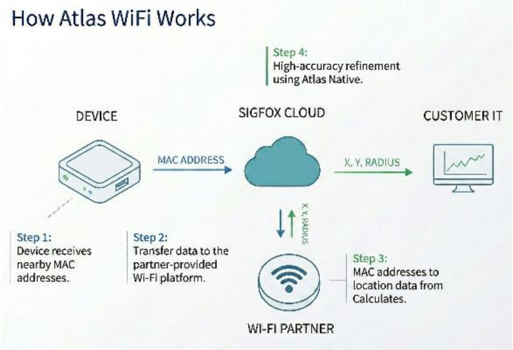


Fig. 5 *Mechanism of Sigfox Atlas WiFi.*

Instead of relying on the existing Atlas WiFi service, this research developed and validated a proprietary mechanism[7] that directly stores MAC addresses acquired from the device into the Sigfox message payload and queries the Here Positioning API through a custom backend server (e.g., AWS). Due to Sigfox communication specifications, the maximum payload for a single message is strictly limited to 12 bytes, allowing for the inclusion of only two MAC addresses (six bytes each). This limitation presents significant challenges:

- **Positioning Failure**: The Here API often cannot calculate a location based on only two MAC addresses, resulting in no coordinate data. An analysis of historical experimental data involving 5,431 unique transmissions revealed that approximately 75.3% of all transmissions failed to acquire even a single valid MAC address (resulting in no data), highlighting the extreme difficulty of data acquisition in sparsely populated areas.
- **Large Errors**: Even when a location is successfully identified, the accuracy can vary significantly, ranging from tens to hundreds of meters when relying on only two data points.
- **Effects of Inappropriate APs**: If mobile access points (APs), such as smartphone tethering devices, or APs with registered locations that differ from their actual physical locations are included in the transmission, the center of gravity for location estimation can be severely distorted.

The following visualization presents experimental data obtained using the conventional single-scan method (maximum of two MACs per location, no MAC filtering). As shown, the estimated coordinate points are widely dispersed across a vast area, visually confirming that practical positioning accuracy has not yet been achieved.

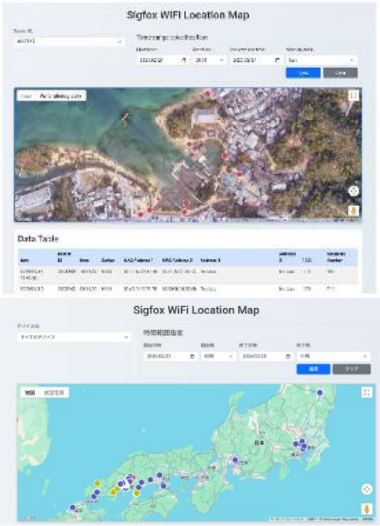


Fig. 6 *Data scattering by conventional methods (max two MACs per location, no MAC filter)*

C. Summary of Challenges

The previous analysis clearly shows that achieving practical accuracy (approximately 10 m, similar to GPS) in custom Sigfox WiFi positioning, while using a minimal number of MAC addresses (two), requires a mechanism to evaluate and prefilter the quality of MAC addresses before transmission.

4. Accuracy improvement method via WiFi MAC address analysis

A. Verification of Continuous Scanning via Walking (Device-side Enhancements)

In this verification, communication and positioning experiments were conducted using a dedicated edge device (shown in Fig. 7) that incorporates an ESP32-S3 chip for WiFi scanning, alongside the aforementioned GPS module.

Fig. 7 *ESP32-S3 edge device*.

Objective: To address payload constraints, the “Continuous Scan” approach, which acquires a maximum of four MAC addresses through overlapping scans in two separate transmissions, was tested for its effectiveness in improving positioning accuracy within a 110-m sectional environment.

Method: Data was collected in a 110-m section (11 points at 10-m intervals) under consistent conditions that simulated stopping every 10 m. Two methods were compared based on the minimum error reported by the Here API: “Single Scan (transmitting a maximum of two MACs per location)” and “Continuous Scan (scanning twice per location to transmit a maximum of four MACs).”

Fig. 8 *110-m section comparison experiment environment*

Summary of Results: In the Single Scan method (two MACs transmitted), the Here API lacked sufficient information to accurately determine the location, resulting in a failure to acquire data at eight of 11 points. At the remaining three points where positioning was successful, the average error was 26.5 m, with a median error of 25.3 m. By contrast, the Continuous Scan method (maximum of four MACs transmitted) successfully determined locations at all 11 points, achieving a 100% acquisition rate. Furthermore, the location information error improved, with an average of 25.8 m and a median of 19.2 m. These results indicate that an accuracy level comparable to standalone GPS can be achieved, and that the dynamic collection of multiple MACs in two stages is an effective strategy for significantly enhancing both the “positioning success rate” and “accuracy” within Sigfox’s severe payload constraints.

Figure 9 illustrates the measurement results for the 110-m section at 10-m intervals. True coordinates are indicated by blue pins, single scan results are shown in orange, and continuous scan results are in red, accompanied by their respective accuracy radii. The single scan (orange) exhibited failures at numerous points, resulting in scattered data, whereas the continuous scan (red) is consistently plotted near the true coordinates (blue) at all points, visually demonstrating its superior tracing capabilities as a trajectory.

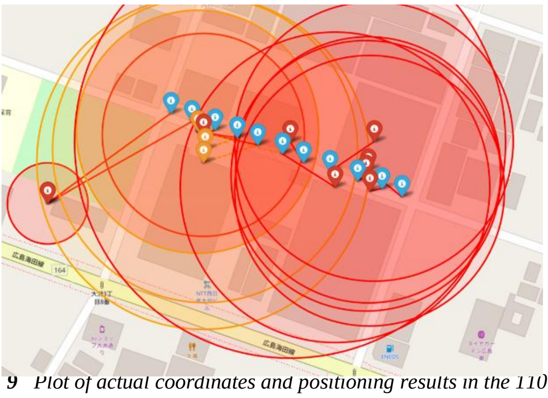
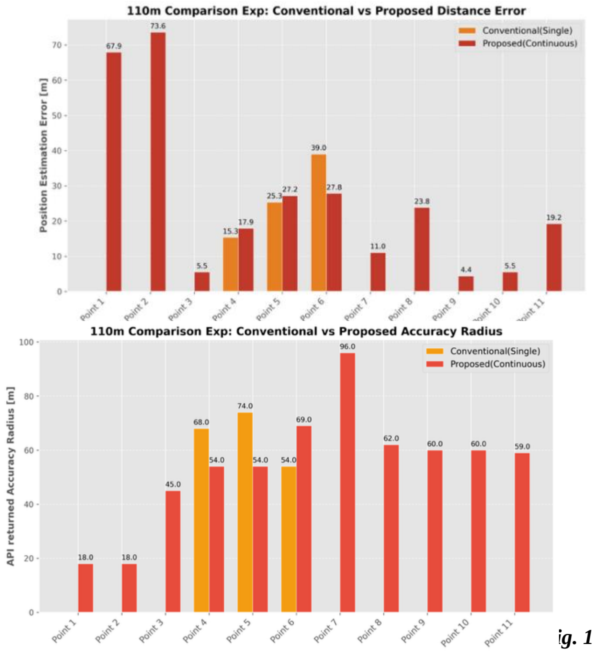


Fig. 9 *Plot of actual coordinates and positioning results in the 110-m section (single vs. continuous).*

Table 1: *presents the actual measured errors for each positioning method across the 110-m section (11 points in total). Entries marked with “-” in the Single Scan column denote points where positioning was not possible, as the Here API did not return coordinate data due to insufficient information.*

Point Name	**Single Scan Error (m)**	**Continuous Scan Error (m)**	**Accuracy Improvement (m) / Notes**
Point 1	-	67.9	Acquisition Improved
Point 2	-	73.6	Acquisition Improved
Point 3	-	5.5	Acquisition Improved
Point 4	15.3	17.9	−2.6
Point 5	25.3	27.2	−1.9
Point 6	39.0	27.8	11.2
Point 7	-	11.0	Acquisition Improved
Point 8	-	23.8	Acquisition Improved
Point 9	-	4.4	Acquisition Improved
Point 10	-	5.5	Acquisition Improved
Point 11	-	19.2	Acquisition Improved
Average	**26.5**	**25.8**	-
Median	**25.3**	**19.2**	-



ig. 10
Comparison of error and accuracy radius under the positioning method

Through the implementation of the aforementioned methodology, which relies on operational ingenuity at the Sigfox device end (Continuous Scanning combined with rough received signal strength indicator (RSSI) filtering), the acquisition rate significantly improved, resulting in a positioning accuracy refined to approximately 19 m.

B. Verification of Effectiveness during Continuous Walking (3 and 6 km/h)

Objective: To assess the feasibility of achieving highly accurate positioning by dynamically collecting and merging four MAC addresses while walking continuously at a constant speed.

Method: Positioning data was gathered within the same 110-m segment while moving at "Low-speed Walking (approximately 3 km/h)" and "High-speed Walking (approximately 6 km/h)." Two sequential messages, each transmitting two MAC addresses, were merged into a single group (with a maximum of four MACs at the backend). The estimated coordinates and accuracy radius were then calculated using the Here API.

Results and Discussion: Table 2 lists the positioning results for each group during continuous walking. When walking at a low speed of approximately 3 km/h, all three groups achieved successful positioning. Notably, Group 1 demonstrated a high precision, namely, a radius of 18 m, which is comparable to standalone GPS positioning. However, a significant issue was noted: when comparing the estimated coordinates (latitude and longitude) to the actual starting point of the walk, Group 1 exhibited a spatial offset exceeding 50 m. This discrepancy was not due to measurement errors from the device; rather, it indicated that the physical locations of the MAC addresses (access points) used by the Here Positioning API did not align with their true real-world positions. Thus, whereas the API reported a location with "high confidence" (an 18-m accuracy radius), inherent biases or deviations in its reference database remained. By contrast, during high-speed walking at approximately 6 km/h, the accuracy radius for Group 1 increased significantly to 89 m. At this speed, equivalent to 100 m per min (approximately 1.67 m/s), the subject covered the entire 110-m section in just 66 s. Given that the subject had already advanced nearly 100 m during the time spent transmitting multiple messages via Sigfox (including scanning intervals), achieving a location "within an 89-m radius" using dynamically acquired MAC addresses represents a sufficiently high and practically effective accuracy considering the speed of movement.

***Table 2:** Positioning results for each group during continuous walking.*

Walking Speed	**Group**	**Transmission Count**	**Estimated Lat**	**Estimated Lng**	**Accuracy Radius(m)**
Approx. 3 km/h	Slow ①	4 MACs	34.38839	132.49108	18
Approx. 3 km/h	Slow ②	4 MACs	34.38852	132.49209	65
Approx. 3 km/h	Slow ③	4 MACs	34.38868	132.49221	68
Approx. 6 km/h	Fast ①	4 MACs	34.38869	132.49235	89

Figure 11 plots the positioning data recorded at each walking speed while continuously scanning along the 110-m route. (The blue line represents the actual 110-m path; green pins indicate Slow Walking; purple pins denote the positioning results and accuracy radius for Fast Walking). The visual representation confirms that data acquisition remains consistently successful even during motion.

Fig. 11 *Positioning results plot during continuous walking (3 and 6 km/h)*

These results validate that the method of merging multiple scans operates effectively while in motion, achieving a reasonable positioning accuracy (ranging from tens of meters to approximately 10 m) relative to walking speed.

To address coordinate fluctuations (discrepancies in accuracy between groups) caused by walking, to achieve a more advanced and consistent improvement (ensuring errors remain under 10 m), and to cope with mobile APs (e.g., smartphone tethering), detailed data accumulation and thorough analysis on the backend (e.g., AWS) are essential. Therefore, to capture nuanced WiFi data that the device alone cannot interpret and to utilize this for constructing whitelists, the custom Here API Validation Tool was developed.

C. Detailed Investigation of the WiFi Environment using NetSpot

To enhance the accuracy of the Here Positioning API, NetSpot was employed to conduct thorough scans of the WiFi environment in the target area. This process effectively gathers the MAC addresses, SSIDs, and RSSI profiles of all nearby access points.

Fig. 12 *NetSpot WiFi environment map, heatmap, SSID list, and time-series graph*

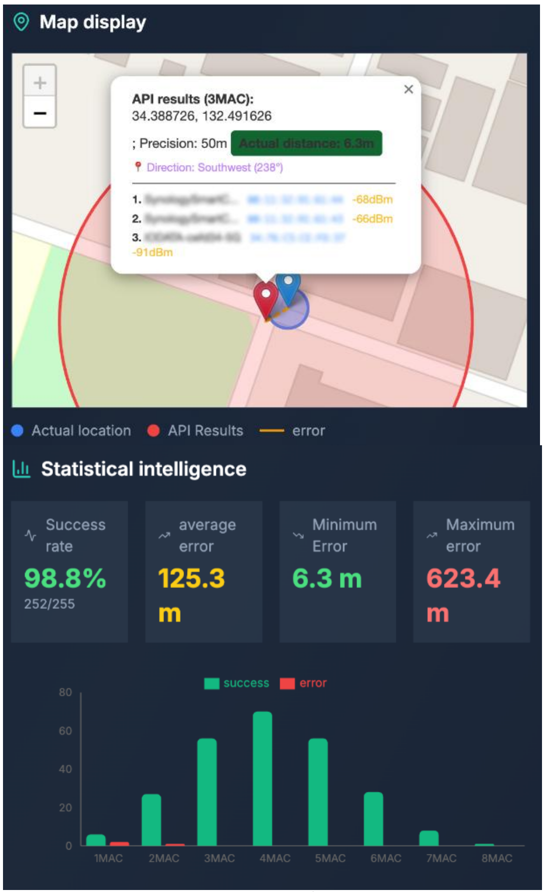
From the data collected from NetSpot, such as the quantitative distribution of APs detected at each location, the localized signal strength patterns, and the temporal stability of APs, the essential baseline data was established for the subsequent analytical phases.

D. Development of the Here API Validation Tool (WiFi Positioning Validator)

To support whitelist registration and advanced accuracy-enhancement measures in the upcoming backend (e.g., AWS), a robust validation tool (WiFi Positioning Validator) was autonomously developed. This tool automatically queries the Here Positioning API with the MAC address sets obtained from NetSpot and calculates the positional error relative to the actual coordinates. The tool includes the following key features:

- Queries the API with various combinations of MAC addresses to determine the coordinate error compared to the ground truth.
- Visualizes the “Contribution Parameter” for each MAC address, indicating how much each AP affects tracking accuracy.
- Analyzes systemic errors in orientation (directional bias vectors).
- Compiles error statistics from various waypoints into comparative analyses.

Through this validation tool, a thorough accuracy audit was conducted. The complete MAC address repository from the NetSpot data, mapped at the same 11 waypoints used in the Continuous Scan experiment, was input into the live Here API. The interface schematics and detailed analytical visualizations for Waypoint 1 are presented in Figs. 13–15.

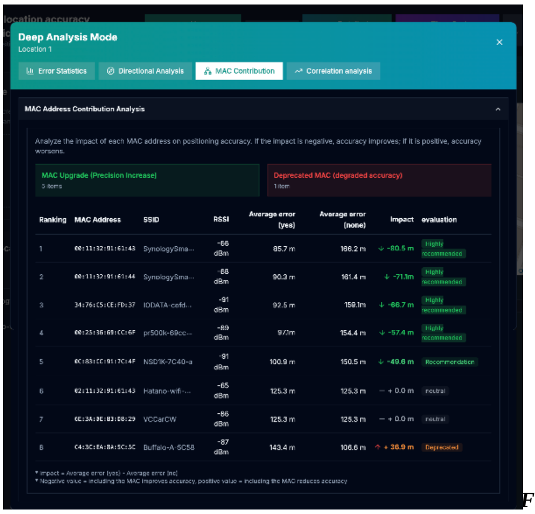


API results (3MAC):
34.388726, 132.491626
; Precision: 50m Actual distance: 6.3m
Direction: Southwest (238°)

1. [redacted] -68dBm
2. [redacted] -66dBm
3. [redacted]
-91dBm

Success rate 98.8% 252/255
average error 125.3 m
Minimum Error 6.3 m
Maximum error 623.4 m

Fig. 13 *Top screen and statistical data of the custom-developed Here API Validation Tool*



MAC Address Contribution Analysis

Analyze the impact of each MAC address on positioning accuracy. If the impact is negative, accuracy improves; if it is positive, accuracy worsens.

MAC Upgrade (Precision Increase)
5 items

Deprecated MAC (degraded accuracy)
1 item

Ranking	MAC Address	SSID	RSSI	Average error (yes)	Average error (none)	Impact	evaluation
1	00:11:32:91:61:43	SynologySma...	-66 dBm	85.7 m	166.2 m	↓ -80.5 m	Highly recommended
2	00:11:32:91:61:44	SynologySma...	-68 dBm	90.3 m	161.4 m	↓ -71.1m	Highly recommended
3	34:76:C5:CE:FD:37	IODATA-cefd...	-91 dBm	92.5 m	159.1m	↓ -66.7 m	Highly recommended
4	00:25:36:69:CC:6F	pr500k-69cc...	-89 dBm	97.1m	154.4 m	↓ -57.4 m	Highly recommended
5	0C:83:CC:91:7C:4F	NSD1K-7C40-a	-91 dBm	100.9 m	150.5 m	↓ -49.6 m	Recommendation
6	02:11:32:91:61:43	Hatano-wifi-...	-65 dBm	125.3 m	125.3 m	— + 0.0 m	neutral
7	6E:3A:0E:83:08:29	VCCarCW	-86 dBm	125.3 m	125.3 m	— + 0.0 m	neutral
8	C4:3C:E4:8A:5C:5C	Buffalo-A-5C58	-87 dBm	143.4 m	106.6 m	↑ + 36.9 m	Deprecated

* Impact = Average error (yes) - Average error (no)
* Negative value = including the MAC improves accuracy, positive value = including the MAC reduces accuracy

Fig. 14 *Analysis of the MAC address contribution to error at Point 1*

Azimuth Heatmap

Displays the direction where errors are likely to occur on a radar chart

north	Northeast	East	Southeast
4 items (2 %) Average 529 m	0 items (0 %) Average 0 m	4 items (2 %) Average 411 m	16 items (6 %) Average 12 m
South	**Southwest**	**West**	**Northwest**
0 items (0 %) Average 0 m	38 items (14 %) Average 81 m	188 items (75 %) Average 134 m	4 items (2 %) Average 89 m

Error vector scatter plot

The error vector of each measurement result is plotted on the XY coordinate. The red diamond indicates the average bias.

Average bias (systematic error)
East-West Offset: 98.7 m west
North-South Offset: 0.4 m north
Average distance: 98.7 m
Main Direction: West (270 °)
* Correction vector: Latitude -0.000003 °, Longitude 0.000899 °

Basic statistics

Error Statistics
Number of data 252 items
average error 125.3 m
Median 125.3 m
Standard deviation 138.6 m
Minimum Error 6.3 m
Maximum error 623.4 m
dispersion 19215.6 m²
range 617.1 m

Error distribution histogram

Fig. 15 *Bearing analysis and detailed statistical graphs at Point 1*

E. Implementation of Proactive Prefiltering on the Device (ESP32-S3)

To improve positioning accuracy at the device level, unique prefiltering and screening architectures were integrated and validated on the edge device (execution environment: ESP32-S3) prior to transmission over Sigfox.

Absolute Exclusion of LAA and Mobile OUIs

- **Mechanism**: During the ESP32-S3 WiFi scanning protocol, randomly generated MAC addresses (LAA: locally administered addresses) and OUIs (vendor codes) associated with mobile devices, which are prone to displacement via tethering, are strictly removed from the MAC cache.
- **Implementation Strategy**: Inspect the first octet of the MAC address to instantaneously identify and discard LAAs (where the second character is 2, 6, A, E). Maintain a dedicated blacklist of mobile OUIs within the ESP32 to continually eliminate MAC addresses originating from mobile terminals that match the blacklist.
- **Empirical Substantiation via the Validation Tool**: The Waypoint 1 audit, conducted with the designated validation tool, conclusively demonstrated that MAC addresses categorized as LAA have no effect on the Here API's positioning calculations (resulting in an error contribution of 0 m). In simpler terms, including these addresses in the limited Sigfox payloads (which allow two to four inputs) constitutes a complete waste of bandwidth. Therefore, a reliable local discard logic on the device side is essential.

Triage Optimization by Apex RSSI

- **Mechanism**: After passing through the filtration process, the most powerful APs, which emit the highest RSSI (signal strength), are prioritized for data transmission.
- **Implementation Evaluation**: Although the ESP32-S3 has sufficient computational bandwidth and memory (SRAM/Flash), the core protocol developed for this validation primarily focuses on applying a simple descending sort based on RSSI values to the filtered array (after LAA/OUI exclusion). Only the top-ranking APs are then used in the payload sequence.

F. Experiment 1: Correlation between MAC Address Volume and Positioning Accuracy

Objective: To quantitatively assess the extreme limits of positioning accuracy defined by the 2-unit Sigfox constraint and to identify the necessary threshold for MAC address inclusion.

Method: The validation tool was used to incrementally increase the volume of MAC addresses injected uniformly across all 11 waypoints, carefully recording and compiling the changes in positioning error.

Summary of Results: Data collected from all 11 waypoints revealed an average error of approximately 112 m when restricted to a 2-unit injection limit, significantly below the expected GPS accuracy of approximately 10 m. Notably, increasing the MAC address payload resulted in a significant improvement in accuracy, peaking at an average error of approximately 36 m, the most stable level of accuracy observed, with the injection of four units. However, raising the throughput threshold to eight units resulted in an average error of approximately 38 m, indicating a plateau (and slight regression) in accuracy improvement. This decline can be attributed to the introduction of distant, weak signal APs (ambient noise) that distort the API's estimation process. This finding underscores that optimal preprocessing logic is not achieved by simply increasing the volume of data but by methodically isolating and querying the optimal echelon of approximately four "superior-quality" MAC networks with maximum broadcast resilience.

G. Experiment 2: Accuracy Elevation through Rigorous MAC Quality Segmentation

Objective: This study aims to evaluate the exact kinetic effects of the "Qualitative Fidelity" of injected MAC addresses on terminal positioning accuracy and to develop optimized selection criteria.

Method: Implement and assess the following three-segment taxonomy to compare raw accuracy differentials.

Segmentation Protocol	Analytical Architecture	Formulated Expectation
RSS Filtration Pivot	Exclusive utilization of top N units displaying paramount RSSI vectors	Primacy applied to hyperstable adjacent arrays
Mobile AP Eradication	Systemic excision of mobile APs classified by chronological manifestation flux models	Eradication of unstable geographical transients
Contribution Diagnostics	Algorithmic calculation of individual MAC liability parameters to ostracize detrimental arrays	Radical deprecation of systemic bias

Summary of Results: Utilizing a foundational RSS filtration baseline (top eight units) combined with rigorous mobile AP excision led to a remarkable increase in accuracy, closely aligning with the 10-m radius target. In addition, applying the "Contribution Diagnostics" protocol from the validation tool to the Waypoint 1 datasets revealed a concerning finding: the integration of specific isolated APs (e.g., C4:3C:EA:...) increased the average coordinate error by a significant 37 m. This serious issue is identified with high certainty as a "corrosive rogue AP," incorrectly logged within the overarching database, and is directly responsible for the "50-m+ locational displacement" documented during the continuous traversal phase near Waypoint 1 in Chapter 4.2. By utilizing this tool, designed specifically to detect and blacklist rogue AP vectors from the operational backend, we can effectively eliminate large-scale systemic deviations from the positioning lifecycle.

H. Experiment 3: Calibration of Systemic Error (Azimuth Bias)

Objective: To create comprehensive error vectors based on azimuth orientation derived from successive positioning outputs and to develop correction matrices that optimize maximum theoretical precision.

Method: Calculate statistical azimuth error vectors from extended multi-point positional outputs to create a dynamic algorithmic bias-correction framework. Assess its overall effectiveness by comparing spatial dispersal matrices before and after calibration.

I. Experiment 4: Spatial Trajectory Filtration Using Chronological Data Clusters

Objective: To address dynamic transit-tracking paradigms by using chronologically ordered positional clusters to eliminate outlier anomalies and enhance tracking precision trajectories.

Method: Conduct iteratively staggered positioning bursts over the Sigfox substrate, then apply trajectory constraints (such as advanced Kalman filters). Analyze whether this temporal matrix yields significantly improved trajectory stabilization compared to isolated point-estimation sequences by incorporating logical movement constraints (e.g., pedestrian velocity limits of ≤ 5 km/h) to physically mitigate erratic positional spikes.

J. Comprehensive Synthesis of Precision Enhancement Frameworks

Based on the empirical findings of this research, the essential methodologies required to overcome the significant Sigfox 2-MAC payload limitations and achieve accuracy levels comparable to GPS paradigms (sub-10-m limits) are outlined as follows:

1. **Device-level Architecture Triage (Continuous Scan Implementation)**: Implement staggered, two-stage sequential scanning clusters combined with macroscopic RSSI truncation at static waypoints to significantly increase MAC throughput (up to four apex-tier inputs), thereby eliminating the high positioning attrition rate seen in single-scan applications and stabilizing coordinate isolation medians at approximately 19 m.
2. **Propulsive Dynamic Scan Integration**: Coupling messages to transmit up to four MAC arrays consistently maintains exceptionally sharp localized accuracy (with minimal extremes at approximately 18 m), matching static orientations, even during movement at speeds of up to 3 km/h.
3. **Backend Algorithmic Tier Segmentation**: Isolate erratic mobile arrays from the device using cloud-based (AWS) whitelisting mechanisms informed by comprehensive WiFi contextual analysis (Here API Validation Tool benchmarks).
4. **Calibration Protocol of Systemic Discrepancies**: Dynamically reduce macro-errors through the azimuth-oriented bias correction framework.
5. **Chronological Filtration Models**: Suppress outlier spikes using dynamic trajectory velocity matrix constraints.

Recently published studies demonstrate that machine learning strategies are improving the accuracy of localized indoor WiFi coordinate estimation to a mean absolute error of approximately 1 m[8]. The advanced filtration techniques and comprehensive modeling approaches discussed here are essential steps toward the next level of location estimation technology.

Integration Potential with the Atlas WiFi Private DB Initiative

In addition to proprietary network applications, a significant opportunity emerges when rigorously verified high-reliability APs are registered in the formal administrative registry of the Sigfox Atlas WiFi Private DB initiative. This could lead to significant improvements in both the overall success rate and accuracy of their official public platform infrastructure.

5. Conclusion

This research addressed the persistent challenges in positioning accuracy caused by the limitations of payload architecture (restricted to two MAC addresses) in custom Sigfox-based WiFi localization systems. The “Continuous Scan” algorithm was introduced as a corrective protocol for devices stationed at fixed waypoints every 10 m. By overlapping multiple transmissions, this protocol enables the dynamic use of up to four MAC addresses, which effectively eliminates the significant failure rate associated with single-scan attempts (which exhibited an approximately 70% failure margin under test conditions) resulting in precise coordinate isolation at each waypoint and stabilizing the overall positional median at approximately 19 m. In addition, empirical tests involving continuous walking at speeds of approximately 3 km/h confirmed that chaining as many as four MAC addresses consistently yields practical and efficient accuracy in dynamic mobile tracking scenarios. This validates the robust functional capability of the system. Critically, data accumulation combined with macro-analytical processing on a centralized backend is vital for achieving greater accuracy levels (maintaining within 10-m thresholds). To this end, we propose a strong MAC address segmentation strategy driven by extensive WiFi environmental analysis using NetSpot, in conjunction with our custom-built Here API Validation Tool. By incorporating the empirical filtering logic validated by this toolset, such as aggressive RSS filtering, elimination of mobile AP profiles, recalibration of systemic biases, and longitudinal tracking models, into the backend network implementations (e.g.,. AWS whitelists) and aligning it with the edge device’s continuous scanning infrastructure, we aim to deliver highly stabilized coordinate projection operations that match or exceed the performance of traditional standalone GPS methodologies.

6. References

[1] B. W. Remondi, “Global positioning system carrier phase: Description and use,” Bulletin Géodésique, vol. 59, no. 4, (1985) pp. 361-377.

[2] U. Raza, P. Kulkarni and M. Sooriyabandara, “Low power wide area networks: An overview,” IEEE Communications Surveys & Tutorials, vol. 19, no. 2, (2017), pp. 855-873

[3] https://docs.arduino.cc/hardware/uno-rev3 (March 1, 2026)

[4] https://unabiz.github.io/unashield/hardware.html (March 1, 2026)

[5] http://akizukidenshi.com/catalog/g/gK-09991/ (March 1, 2026)

[6] T. Tanaka, N. Okamitsu, and K. Nishino, IoT device with Global Positioning System using Sigfox, ThingSpeak, and Anaconda platform, E+E, vol. 55, (2020)pp. 94-98.

[7] M. Watanabe, S. Ohji, T. Ishikawa, Y. Ueda, S. Tada, R. Masumoto, M. Yamauchi, K. Kakugawa, T. Tanaka, and K. Vutova, “Location estimation using Wi-Fi media access control address and device miniaturization”, Industry 4.0, vol. 10, no. 2, (2025) pp. 43-46.

[8] H. Esmaeili Gorjan and V. P. Gil Jiménez, “Improving indoor WiFi localization by using machine learning techniques,” Sensors, vol. 24, no. 19, (2024) p. 6293.