Enhancing LoRaWAN Communication for Mobile Nodes with Techniques for Predicting Signal Strength

Hristijan Slavkoski[1,2], Simeon Trendov[1], Eduard Siemens[1] and Marija Kalendar[2]
[1]*Department of Electrical, Mechanical and Industrial Engineering, Anhalt University of Applied Sciences, Bernburger Str. 55, 06366 Köthen, Germany*
[2]*Faculty of Electrical Engineering and Information Technologies, SS. Cyril and Methodius University, Rugjer Boshkovik Str. 18, 1000 Skopje, North Macedonia*
hristijan.slavkoski@student.hs-anhalt.de, {simeon.trendov, eduard.siemens}@hs-anhalt.de, marijaka@feit.ukim.edu.mk

Keywords: Adaptive Control, Adaptive Data Rate (ADR), Coding Rate (CR), Energy Efficiency, Internet of Things (IoT), Kalman Filter, LoRaWAN, Mobile Communication, Received Signal Strength Indicator (RSSI), Signal-to-Noise Ratio (SNR), Signal Strength Prediction, Spreading Factor (SF), Transmit Power (TP).

Abstract:

Mobile LoRaWAN links suffer from rapid RSSI/SNR fluctuations due to motion, obstacles, and interference, degrading reliability and wasting energy. This work evaluates lightweight signal-strength prediction combined with adaptive control on resource-constrained hardware. A Kalman filter is applied to smooth per-packet RSSI/SNR and to trigger parameter updates to transmit power, spreading factor, and coding rate only when persistent degradation is detected. The approach is implemented on an Arduino sender and a Raspberry Pi receiver and tested in urban, rural, park, and free-field environments. Results show variance reductions in RSSI of about one third and SNR of about one fifth, translating into energy savings of 15–27% without loss of reliability. Compared with a reactive baseline and the principles of LoRaWAN ADR, the method responds faster to recovery and avoids prolonged high-power operation in mobility. The findings indicate that simple predictive filtering is an effective building block for robust and energy-efficient mobile LoRaWAN systems.

1 INTRODUCTION

The IoT relies on efficient and reliable communication, often in remote or mobile scenarios. Among LPWAN protocols, LoRaWAN stands out for its long range and energy efficiency, making it ideal for battery-powered devices in rural or infrastructure-scarce areas. However, in mobile conditions, signal strength varies unpredictably due to distance, interference, obstacles, and motion, causing packet loss, latency, and unstable connectivity.

This work explores lightweight methods for predicting signal strength in mobile LoRaWAN nodes to stabilize links and improve energy efficiency. By forecasting link quality, the system can adjust transmission parameters or delay non-urgent messages until conditions improve. The approach combines LoRaWAN's low-power design with simple predictive models, such as the Kalman filter, implemented on microcontrollers and single-board computers like Arduino or Raspberry Pi. The following sections outline the hardware setup, system architecture, and prediction techniques, followed by results and future work.

2 SYSTEM DESIGN AND IMPLEMENTATION

To evaluate signal prediction and improvement methods in mobile scenarios, a compact IoT system was developed with two components: a mobile sender node that transmits packets while moving, and a stationary receiver that collects, filters, and analyzes signal data. The aim was to create a lightweight, reproducible setup capable of capturing real-time signal variations with minimal hardware complexity.

2.1 Hardware Setup

The hardware architecture is designed to mimic practical IoT deployments where cost, portability, and energy efficiency are critical. Both nodes use commercially available components to ensure reproducibility and scalability.

The sender node (Fig. 1) is based on an Arduino Mega 2560 with a Dragino LoRa Shield v1.4 [1] using the Semtech SX1276 [2] transceiver operating at 868 MHz. It is powered by a 5 V power bank and

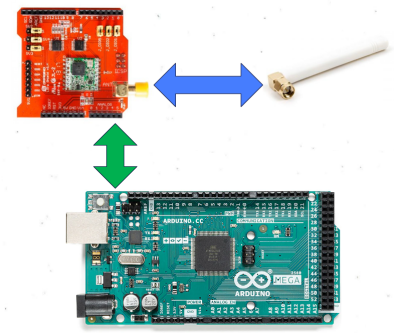

Figure 1: Mobile sender node based on Arduino Mega with Dragino LoRa Shield.

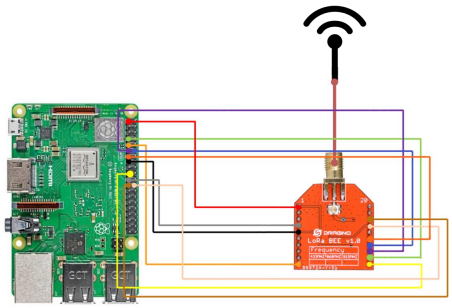

Figure 2: Stationary receiver node with Raspberry Pi 3 B+ and Dragino LoRa BEE.

uses an SMA omnidirectional antenna. The stationary gateway (Fig. 2) employs a Raspberry Pi 3 B+ connected to a Dragino LoRa Bee v1.1 (SX1276, 868 MHz) [3] through SPI, using the same type of antenna for compatibility. This configuration reflects a practical IoT deployment: portable, energy-efficient, and built entirely from off-the-shelf components.

2.2 Software Architecture

The Arduino runs C++ firmware that periodically sends numbered packets and waits for acknowledgments (ACKs) containing updated communication parameters. The receiver, implemented in Python on the Raspberry Pi, listens for packets, logs the RSSI and SNR, and applies a lightweight Kalman filter to smooth out fluctuations caused by movement and interference.

Based on the filtered RSSI and SNR values, the receiver adaptively suggests adjustments to the sender's TP, SF, or CR. These updated parameters are embedded within standard ACK messages, ensuring synchronization without additional overhead. If packet ACKs are missed, the sender increases its TP until the link stabilizes. All packets and adaptation events are timestamped and stored for later analysis.

This feedback loop enables real-time link optimization directly on low-power hardware. The simplicity of the implementation shows that predictive control can run on resource-limited devices such as Arduino and Raspberry Pi, offering an accessible base for further experimentation.

3 SIGNAL STRENGTH PREDICTION

This section outlines the methodology for managing and interpreting signal strength data in mobile LoRaWAN nodes within an adaptive communication system. The core technique is the Kalman filter, a lightweight algorithm that smooths noisy measurements to improve decision-making. The section explains the filter's mechanics, its application to RSSI and SNR smoothing, and its main limitations and future improvement prospects [4].

3.1 Kalman Filter Overview

Originally developed for navigation, the Kalman filter estimates a system's state by combining predictions with new measurements [5], [6]. Here, the state represents the underlying RSSI or SNR, while noise arises from interference, distance, and multipath effects. The recursive prediction–correction process tracks real changes while suppressing short-term fluctuations.

3.1.1 Prediction, Correction, and Smoothing

The Kalman filter operates in two iterative steps. During the prediction step, it estimates the next RSSI or SNR value from the current state, assuming gradual change with process noise (Q) representing expected variation between measurements. The uncertainty of this prediction is tracked through a covariance matrix. When a new measurement arrives, the correction step combines it with the prediction, weighted by measurement noise (R), to obtain an updated estimate with reduced uncertainty. The Kalman gain (K) determines how much confidence to assign to the new measurement versus the model.

This cycle repeats for every received packet, providing a smoothed, adaptive estimate of link quality. In mobile LoRaWAN systems, this smoothing is crucial: RSSI and SNR often fluctuate sharply due to reflections, interference, or brief obstructions. Without filtering, such transient drops could trigger unnecessary transmission power increases. The Kalman filter instead highlights persistent signal trends while attenuating random noise, enabling the control logic to respond only to meaningful degradation.

3.1.2 Implementation Details

Separate filters were applied to RSSI and SNR, initialized at –90 dBm and 0 dB with tuned parameters:

- Process noise (Q) defines expected signal change. Smaller Q (0.01) stabilizes walking scenarios, while larger Q (0.1) adapts faster to vehicular motion.
- Measurement noise (R) represents sample variance, for example $R = 1.5$. A higher R makes the filter less sensitive to random fluctuations.

For slow movement, small Q and moderate R provided stable estimates. Increasing Q improved responsiveness at higher speeds. Packets were transmitted every five seconds, which was sufficient for walking while keeping computation light.

Filtered outputs were then passed to the adaptive logic. Instead of reacting to every fluctuation, the controller used two thresholds and a dwell period requiring several consecutive packets before adjusting parameters. When filtered RSSI or SNR dropped below the lower limits, TP, SF, or CR increased; when they rose above, parameters were gradually reduced to save energy.

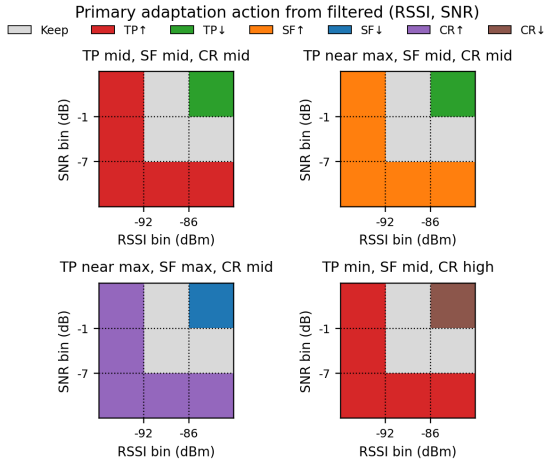

Figure 3: Decision heatmap.

This behavior approximates discrete minimization of the cost function

$$J(p) = E(\mathrm{TP}, \mathrm{SF}, \mathrm{CR}) + \lambda \sum_{i \in \{RSSI, SNR\}} \mathbb{I}\left(\hat{\gamma}_i < \theta_i^-\right), \tag{1}$$

where $p \in \{0,\ldots,12\} \times \{7,\ldots,12\} \times \{\frac{4}{5},\ldots,\frac{4}{8}\}$ is the parameter vector, $E(\cdot)$ denotes energy per packet, $\mathbb{I}(\cdot)$ is the indicator function, θ_i^- are lower thresholds (e.g., $\theta_{RSSI}^- = -92$ dBm), and λ weights reliability against energy. Due to hardware limits, this optimization is implemented as a greedy threshold policy rather than solved explicitly. Equation (1) formalizes the trade-off between minimizing energy and maintaining reliable link quality.

Figure 3 shows the decision space over filtered RSSI–SNR bins. The combination of smoothing, hysteresis, and dwell time prevents frequent toggling, ensuring a stable and energy-efficient control loop.

4 EXPERIMENTAL PROTOCOL AND DATA COLLECTION

To evaluate prediction and adaptation under realistic conditions, experiments were conducted in four environments: urban, rural, park, and free field. Each setting exposed the system to distinct propagation challenges involving interference, obstacles, and open-line attenuation [7], [8], [9]. Instead of maximizing packet counts, the goal was to capture diverse real-world mobile LoRaWAN behavior.

4.1 Field Environments and Motion Profiles

The four test sites (Fig. 4) captured varied propagation conditions. The urban area featured frequent line-of-sight blockages from pedestrians and buildings, causing rapid RSSI/SNR fluctuations. The rural route followed a long village road with minimal interference and gradual distance-driven attenuation. The park offered semi-open terrain with few obstructions, serving as a clean baseline for short-term variability. The free-field site, located on an elevated open area, provided near-ideal line-of-sight conditions isolating fundamental signal dynamics.

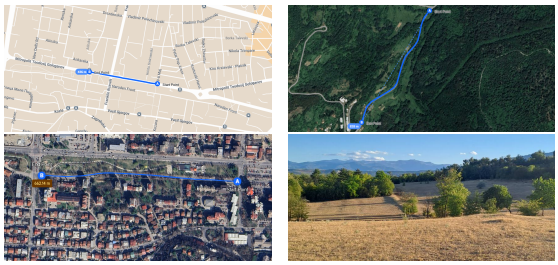

Figure 4: Measurement environments: urban (top left), rural (top right), park (bottom left), and open field (bottom right).

Across all environments, two motion profiles were tested: slow walking at 4–5 km/h and faster traverses at 9–10 km/h, enabling evaluation of the Kalman filter's responsiveness under differing mobility conditions.

4.2 Experimental Setup and Data Logging

The sender used default LoRa parameters and transmitted numbered packets at a fixed cadence. Each packet carried a CRC for integrity, while the receiver logged both raw and filtered RSSI/SNR values. When thresholds were crossed, adaptive commands adjusted TP, SF, or CR. This produced two datasets: a raw (unfiltered) control path and a filtered adaptive path, enabling direct comparison.

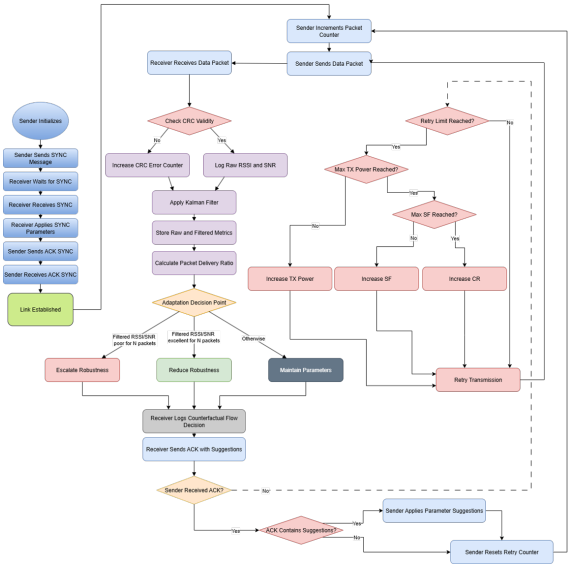

Figure 5: Adaptive packet flow and decision loop between sender and receiver.

The complete communication flow is shown in Figure 5. Each received packet triggered the computation of RSSI/SNR statistics, Kalman filtering, and rule-based adaptation, followed by an ACK containing updated parameters. All log entries included timestamps, signal metrics, and adaptation decisions, allowing precise reconstruction of the decision process and evaluation of filter behavior.

To quantify motion intensity, a simple mobility indicator was derived from the median absolute derivative of filtered RSSI over time, $\text{median}\left(\left|\frac{d\,\text{RSSI}_f}{dt}\right|\right)$, expressed in dB/s. Higher values indicated faster movement or sudden environmental transitions.

4.3 Representative Dataset

From the complete dataset, four representative traces, one from each environment, were selected for detailed analysis. The urban case highlighted sharp, short-lived fades caused by pedestrians, largely suppressed by filtering. The rural case showed smooth distance-driven decline accurately tracked by the filter. The park and free-field results confirmed baseline behavior under clean propagation, revealing how the Kalman filter maintains stability without masking true signal trends. These examples collectively demonstrate how the adaptive control loop performs under varying mobility and environmental complexity.

5 RESULTS AND ANALYSIS

Building on the experimental setups described previously, this section evaluates how the prediction and control loop behaved under real mobility across urban, rural, park, and free-field traces. The analysis focuses on reliability, stability, responsiveness to speed, and energy efficiency using time-series comparisons, variance summaries, and paired slow/fast traverses.

5.1 Reliability Outcomes

Across all runs, end-to-end reliability remained high. Packets were received with valid CRCs and acknowledgments were consistently returned. In the urban trace, temporary fades appeared when pedestrians briefly obstructed the line-of-sight. Such events could trigger false alarms in a threshold-only controller, but the Kalman filter smoothed them effectively, allowing the system to avoid unnecessary power increases.

In the rural trace, RSSI declined gradually with distance, and the adaptation policy reacted only to persistent degradation, confirming correct detection even under smoothing.

5.2 Smoothing and Stability

Stability was evaluated by comparing the variance of raw and filtered RSSI/SNR values across environments. High variance implies frequent threshold crossings that lead to erratic control. By reducing variance, the Kalman filter allowed the controller to focus on persistent changes rather than noise.

In both urban and rural traces, the filter reduced RSSI variance by roughly one-third and SNR variance by about one-fifth. These reductions yielded steadier control; most short fades were removed while the long-term trend remained. In the rural and park tests, where path loss dominated, the filter preserved the overall trajectory but damped small oscillations caused by reflections.

Figures 6 and 7 illustrate these effects: raw traces fluctuate around thresholds, while filtered curves remain stable. The filter thus balanced smoothness and

responsiveness, improving control reliability without masking genuine degradation.

This balance is critical. An adaptive system that chases every spike will waste energy and become unstable, while one that smooths too aggressively may miss the onset of real problems. By achieving less jitter while still tracking the overall decline, the Kalman filter demonstrated that stability and responsiveness can coexist. These results show that smoothing is not just cosmetic but an essential component of reliable control in mobile LoRaWAN links.

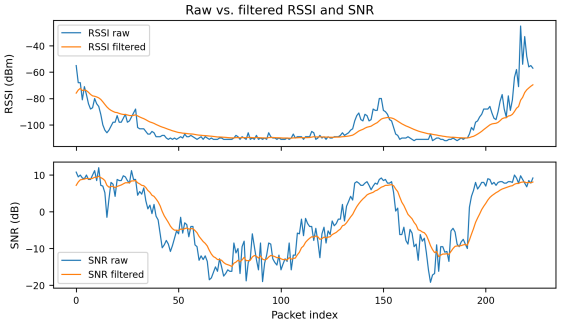

Figure 6: Time-series of raw and filtered RSSI and SNR.

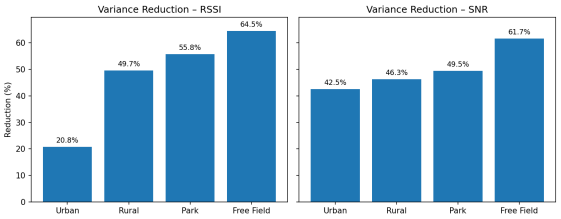

Figure 7: Variance reduction across environments.

5.3 Slow vs. Fast Motion Sensitivity

The system's responsiveness depends on how quickly channel conditions change. Two runs of equal distance but different speeds were compared: a slow walk (4–5 km/h) and a fast traverse (9–10 km/h).

In the slow run, the filtered RSSI followed the trend closely, suppressing minor dips. In the fast run, it lagged slightly behind rapid losses because the process noise Q was conservatively tuned. The link remained stable, but the delay showed that parameters were not optimal for higher speeds.

This trade-off illustrates a general rule: small Q favors stability, larger Q faster adaptation at the cost of more noise. A refinement would be to adjust Q dynamically with estimated speed, while hysteresis and dwell thresholds prevent false triggers. Figure 8 shows consistent control decisions across speeds.

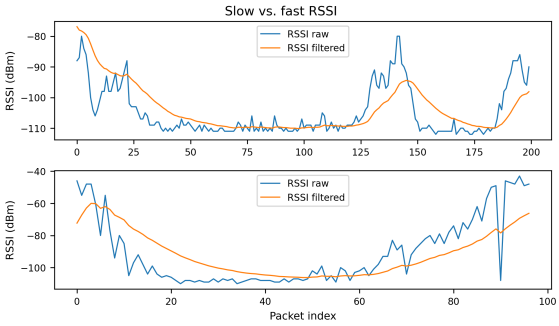

Figure 8: Comparison of slow and fast traverses.

5.4 Raw vs. Filtered Paths Decisions

Because both raw and filtered decision paths were logged, it was possible to compare what the controller would have done without filtering against what it actually did.

In the urban run, the raw path triggered repeated power increases from short fades, while the filtered path stayed steady. In rural and park runs, where signal changes were gradual, both paths mostly agreed, showing that smoothing preserved real degradations.

Filtering thus reshaped the decision process: it reduced impulsive reactions and made control more selective, adjusting parameters only after sustained poor link quality. Dwell counters or hysteresis rules can further confirm persistence before action, improving efficiency without compromising reliability.

5.5 Energy Impact and ADR Comparison

The final analysis quantifies how predictive filtering improved energy efficiency compared with a purely reactive controller and standard LoRaWAN ADR.

5.5.1 Methodology for Energy Measurement

Energy per packet was estimated using Time-on-Air (ToA) and transmit current (I_{tx}) values obtained from LoRaWAN transmission models [10, 11, 12, 9]. Based on the logged transmission sequences, total energy consumption was calculated for two control paths: adaptive (filtered) and reactive (raw). The relative energy savings were then computed as shown in (2), enabling a direct comparison between predictive and reactive control under identical experimental con-ditions.

$$E_{\text{saved}}(\%) = 100\left(1 - \frac{E_{\text{adaptive}}}{E_{\text{reactive}}}\right), \qquad (2)$$

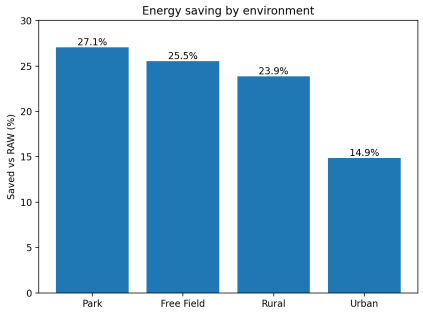

Figure 9: Energy savings by environment.

5.5.2 Quantified Savings and Comparison

The results in Figure 9 show consistent improvements across all environments. Energy use dropped by approximately 27% in the park, 25% in free-field, 24% in rural, and 15% in urban conditions. Savings were highest where transient fades were common, since filtering prevented over-reactions.

A comparison was made with the standard LoRaWAN ADR strategy. ADR is primarily designed for stationary devices and focuses on long-term stability, which often leads to conservative behavior after a degradation event [13]. It typically maintains high transmission power for extended periods, even when conditions have already improved.

The adaptive method evaluated here was tuned for mobility. It increases parameters only after sustained evidence of poor link quality and quickly returns to lower power once conditions recover. This agility allows it to maintain reliability while avoiding unnecessary energy expenditure, achieving a lower power footprint than traditional ADR systems [14].

Overall, predictive filtering shifted control from a reactive to a proactive mode. By distinguishing between noise and genuine degradation, it preserved communication reliability while significantly reducing energy consumption, which is critical for mobile IoT devices operating under strict power constraints.

6 DISCUSSION AND FUTURE WORK

The results show that prediction and smoothing significantly improve the stability of mobile LoRaWAN links under fading and motion. However, several limitations remain, and there are clear opportunities to extend the system. This section reflects on practical constraints and outlines directions for future research.

The Kalman filter's effectiveness depends on parameter tuning, especially the process noise Q and measurement noise R. A conservative Q ensured stability during slow motion but caused lag at higher speeds. A single fixed parameter set cannot perform well under all mobility conditions, suggesting that machine-learning-based tuning could enhance adaptability [15]. The dataset, though covering four environments (urban, rural, park, free field), omits vehicular and indoor scenarios, which would reveal additional fading effects and test generality.

The sender–receiver setup (Arduino Mega and Raspberry Pi 3 B+ with SX127x) ensured reproducibility but limits generalization. Other radios, antennas, or stacks may behave differently, while high delivery ratios partly reflect controlled conditions.

Finally, the optimization targeted reliability, reducing power only under strong link quality. Real deployments may require balancing lifetime or interference instead. Multi-objective control remains challenging for large-scale LoRaWAN systems [16].

Overall, the prototype stabilizes decisions and avoids overreaction but does not yet provide a universal solution for all mobility profiles or device types.

7 CONCLUSIONS

This work demonstrated that predictive filtering combined with adaptive control significantly improves the reliability and energy efficiency of mobile LoRaWAN communication. Field experiments across urban, rural, park, and free-field environments showed that lightweight filtering, such as the Kalman approach, effectively stabilizes fluctuating RSSI and SNR values under mobility, providing a dependable foundation for transmission control.

Filtering cut variability by one-third and saved 15–27% energy without affecting reliability. By suppressing short-lived fades and preventing unnecessary power escalation, the system maintained high packet delivery ratios while conserving energy. Compared with standard ADR, the adaptive approach responded faster to link recovery, restoring efficiency that ADR often leaves unused in mobile scenarios.

However, several limitations remain. The Kalman filter's performance depends heavily on parameter tuning, and a fixed setup cannot optimally handle all motion regimes. The dataset excluded key scenarios such as vehicular or indoor mobility and transitions between line-of-sight and non-line-of-sight. The Arduino–Raspberry Pi prototype ensured reproducibility but limits generalization to other radios or denser deployments. Moreover, the control objective was restricted to maximizing reliability rather than jointly optimizing lifetime, interference, and throughput.

Future work should explore adaptive parameter tuning, multi-objective optimization, and context-aware control using GPS or inertial data. Combining filtering with machine learning - e.g., predictive models or reinforcement learning - could enable autonomous adaptation. Scaling to multi-node networks will further test scalability and coordination.

The findings confirm that intelligent signal processing is a key enabler for robust and sustainable mobile LoRaWAN systems, paving the way for dependable IoT communication under dynamic conditions.

ACKNOWLEDGEMENT

We acknowledge support by the German Research Foundation (Deutsche Forschungsgemeinschaft, DFG) and the Open Access Publishing Fund of Anhalt University of Applied Sciences.

REFERENCES

[1] Dragino, "Lora shield: Long range wireless transceiver for arduino - datasheet," 2016.

[2] Semtech Corporation, SX1276/77/78/79 – 137 MHz to 1020 MHz Low Power Long Range Transceiver, rev. 5 ed., Aug. 2016.

[3] Dragino, LoRa GPS HAT — User Manual, Mar. 2019.

[4] I. Aqeel, E. Iorkyase, and H. Zangoti, "Lorawan-implemented node localisation based on received signal strength indicator," IET Wireless Sensor Systems, vol. 13, pp. 117–132, 2022, doi: 10.1049/wss2.12039.

[5] H. Vo, V. H. L. Nguyen, and V. L. Tran, "Advance path loss model for distance estimation using lorawan network's received signal strength indicator (rssi)," IEEE Access, vol. 12, pp. 83 205–83 216, 2024, doi: 10.1109/ACCESS.2024.3412849.

[6] F. Zafari, A. Gkelias, and K. K. Leung, "A survey of indoor localization systems and technologies," IEEE Communications Surveys & Tutorials, vol. 21, no. 3, pp. 2568–2599, 2019, doi: 10.1109/COMST.2019.2911558.

[7] J. Hérard, L. Odorico, and V. Chauvin, "Flex2trl: A study tool to evaluate the lorawan covering range based on rssi," in Arai, K. (eds) Proceedings of the Future Technologies Conference (FTC) 2024, ser. Lecture Notes in Networks and Systems. Cham: Springer, 2024, vol. 1154, pp. 127–158, doi: 10.1007/978-3-031-73110-5_34.

[8] F. Lemic, A. Behboodi, and J. Famaey, "Location-based discovery and vertical handover in heterogeneous low-power wide-area networks," IEEE Internet of Things Journal, vol. 6, no. 6, pp. 10 150–10 165, Dec. 2019, doi: 10.1109/JIOT.2019.2935804.

[9] S. Trendov, E. Sariiev, and K. B. S. Mukhtar, "Comparison of performance and power consumption in sigfox, nb-iot, and lte-m," in Applied Innovations in Information and Communication Technology. ICAIIT 2024, ser. Lecture Notes in Networks and Systems, S. Dovgyi, E. Siemens, L. Globa, O. Kopiika, and O. Stryzhak, Eds. Cham: Springer, 2024, vol. 1338, pp. 127–158, doi: 10.1007/978-3-031-89296-7_8.

[10] C. Savithi and C. Kaewta, "Multi-objective optimization of gateway location selection in long-range wide area networks: A tradeoff analysis between system costs and bitrate maximization," Journal of Sensor and Actuator Networks, vol. 13, no. 1, p. 3, 2024.

[11] H. Kuchuk, I. Chumachenko, N. Marchenko, N. Kuchuk, and D. Lysytsia, "Method for calculating the number of iot sensors in environmental monitoring systems," Advanced Information Systems, vol. 9, no. 3, pp. 66–72, 2025.

[12] Semtech Corporation, LoRa Modem Designer's Guide, rev. 2 ed., 2019.

[13] S. Trendov, E. Stoilkovska, and E. Siemens, "Impact of LoRaWAN operational parameters on energy efficiency and ways to improve it," in Information and Communication Technologies and Sustainable Development (ICT&SD 2022), ser. Lecture Notes in Networks and Systems, S. Dovgyi, O. Trofymchuk, V. Ustimenko, and L. Globa, Eds. Cham: Springer, 2023, vol. 809, doi: 10.1007/978-3-031-46880-3_10.

[14] A. Augustin, J. Yi, and T. Clausen, "A study of lora: Long range & low power networks for the internet of things," Sensors, vol. 16, no. 9, p. 1466, Sep. 2016, doi: 10.3390/s16091466.

[15] K. Z. Islam, D. Murray, and D. Diepeveen, "Machine learning-based lora localisation using multiple received signal features," IET Wireless Sensor Systems, vol. 13, no. 4, pp. 133–150, Jun. 2023, doi: 10.1049/wss2.12063.

[16] M. Jouhari, N. Saeed, and M.-S. Alouini, "A survey on scalable lorawan for massive iot: Recent advances, potentials, and challenges," IEEE Communications Surveys & Tutorials, vol. 25, no. 3, pp. 1841–1876, 2023, doi: 10.1109/COMST.2023.3274934.