Research Square

Preprints are preliminary reports that have not undergone peer review.
They should not be considered conclusive, used to inform clinical practice,
or referenced by the media as validated information.

Design and Hardware Performance of a LoRa-Based Vehicle Tracking System with Local Data Storage Infrastructure

Alejandro H. Espera

ahesperajr@addu.edu.ph

Ateneo de Davao University

Jenith L. Banluta

Ateneo de Davao University

Princess Camille L. Abar

Ateneo de Davao University

Gabrielle John N. Undangan

Ateneo de Davao University

April M. Salazar

Ateneo de Davao University

Article

Keywords: LoRa, IoT Hardware, Vehicle Tracking, Local Database, Antenna Design, Packet Delivery Rate, RSSI, TDMA

Posted Date: July 3rd, 2026

DOI: https://doi.org/10.21203/rs.3.rs-9618739/v1

License: This work is licensed under a Creative Commons Attribution 4.0 International License.
Read Full License

Additional Declarations: No competing interests reported.

Abstract

This paper presents the design and performance evaluation of a LoRa-based vehicle tracking system, emphasizing its hardware architecture and field performance. The system uses custom-built end nodes equipped with Heltec WiFi LoRa 32 (V3) microcontroller boards (integrating an ESP32 microcontroller and SX1262 LoRa transceiver) and NEO-6M GPS modules, which transmit real-time location data to a centrally located LoRa receiver node. Unlike typical cloud-integrated LoRaWAN solutions, this network operates in a standalone point-to-point topology with a local database server, thereby eliminating the need for external network infrastructure. We detail the hardware integration of the LoRa radio, GPS, and antenna design, including a compact enclosure for in-vehicle units and a high-placement antenna for the central receiver. Multiple transmission schemes were implemented to coordinate multiple vehicles: fixed-interval broadcasting, randomized interval transmissions, and a beacon-synchronized TDMA protocol for collision avoidance. Field tests were conducted in both a suburban open road and a dense urban downtown area to benchmark range, signal quality, and packet delivery reliability under varied propagation conditions. The hardware achieved robust long-range communication up to ~ 4.5 km in suburban line-of-sight conditions, while urban coverage reached approximately 1.0–1.5 km due to building obstructions. The measured Received Signal Strength Indicator (RSSI) levels and Packet Delivery Rate (PDR) are analyzed for each environment: in suburban trials RSSI remained above − 100 dBm up to ~ 700 m and PDR averaged ~ 85–90%, whereas in urban trials RSSI fell off faster (e.g. -107 dBm at 700 m) and PDR ranged ~ 70–80%, with occasional drops to ~ 64% in the most obstructed scenarios. The results demonstrate that a bespoke LoRa hardware platform with local data handling can serve as a low-power, cost-effective alternative to cellular or LoRaWAN-based trackers for vehicular applications. We discuss hardware-oriented optimizations, including antenna elevation and transmission scheduling, which improved reliability. Overall, the proposed system demonstrates the feasibility of a standalone LoRa network for real-time vehicle tracking while highlighting remaining challenges in urban RF performance and scalability across multiple nodes.

1. Introduction

The rise of the Internet of Things (IoT) has enabled new approaches to vehicle tracking and fleet management using wireless sensor networks. Traditional GPS-based tracking systems often rely on cellular networks or satellite communication to relay location data, which incurs high operational costs and requires ubiquitous network coverage. As an alternative, Low-Power Wide-Area Network (LPWAN) technologies such as LoRa (Long Range) have attracted attention for providing kilometer-range wireless links with very low power consumption [1, 2]. LoRa operates in unlicensed sub-GHz bands (e.g., 433 MHz, 868 MHz, 915 MHz) and employs Chirp Spread Spectrum modulation, achieving reliable connectivity over distances of several kilometers even in obstacle-dense environments, while consuming minimal energy [3]. These attributes make LoRa particularly attractive for IoT applications, such as asset tracking and smart city sensors, that require long-range communication on battery-powered devices [3, 10]. These characteristics have been extensively analyzed in foundational LoRa studies [5, 11]. Operation within the 915–918 MHz ISM band complies with Philippine spectrum allocation guidelines [15].

Most LoRa deployments leverage the LoRaWAN protocol, which introduces a cloud-connected network server and gateway infrastructure to manage many end devices in a star-of-stars topology [3]. However, LoRaWAN's dependence on internet-connected gateways and a centralized server can be a drawback in regions with underdeveloped connectivity or for projects requiring complete local control of data [3]. In the Philippines' transportation sector, connectivity dead zones and high subscription costs pose challenges to continuous vehicle tracking using cellular or LoRaWAN networks. To address these gaps, we explore a *standalone LoRa-based network* for vehicle tracking, in which LoRa transceivers communicate point-to-point and log data to a local database without relying on external networks. By adopting a localized architecture, the system aims to provide real-time tracking with low recurring costs, improved data privacy, and resiliency against internet outages. The protocol operation and network architecture are formally defined by the LoRa Alliance specification [6].

Several recent studies have highlighted the feasibility of LoRa for vehicle and asset tracking. For example, de Camargo *et al.* evaluated LoRaWAN trackers on garbage trucks in a smart city context. They found that a custom ESP32 LoRa with GPS device (TTGO T-Beam) outperformed off-the-shelf trackers, reliably sending coordinates without being hindered by vehicle motion or interference [4, 10]. Their experiments reported effective communication ranges of a few kilometers in urban environments when using high gateway placements, and nearly 100% packet reception in areas of good coverage [4]. Other work has shown that LoRa links can achieve up to 3–5 km in urban settings and 12–15 km in rural open areas under ideal conditions [1, 2]. These ranges can extend even further (over 30 km) with elevated base station antennas in rural line-of-sight scenarios. Such performance demonstrates LoRa's potential to cover city-wide areas with only a handful of base stations, albeit at the cost of throughput and capacity trade-offs typical of LPWAN technology.

Adapting LoRa to real-time vehicle tracking presents unique hardware and networking challenges. Unlike static sensor networks, a vehicular tracker must handle rapid changes in position and signal conditions, as well as concurrent transmissions from multiple vehicles. The vehicle's hardware needs to be compact, lightweight, and rugged, integrating a GPS receiver, a LoRa radio, a microcontroller, and a battery/power supply in an enclosure that can be mounted with a clear line of sight for GPS. Antenna selection and placement are critical for maximizing range – typically, a vertically polarized 915 MHz monopole is used, and the receiving gateway antenna should be elevated to avoid ground-clutter attenuation [4]. Furthermore, while LoRa's ALOHA-based media access in simple point-to-point links is extremely simple, it risks packet collisions when many nodes transmit unsynchronized. Custom Medium Access Control (MAC) strategies (such as listen-before-talk, scheduled time slots, or frequency hopping) are often needed to ensure reliability in multi-node LoRa networks [3].

In this paper, we build on these insights to develop a LoRa-based hardware platform for vehicle tracking with an emphasis on the engineering of its components and performance characterization. The system's novelty lies in its fully localized network design and hardware-centric optimizations. We detail the system architecture, consisting of multiple vehicle-mounted LoRa/GPS tracker units and a single central LoRa receiver node interfaced to a local database server, and describe the embedded hardware integration of

the LoRa radio (Semtech SX1262) with the microcontroller and GPS. Special focus is given to antenna design and placement for both the transmitter and receiver, as well as to the packaging of the electronics into a robust, field-deployable enclosure. We also present the transmission control methodologies implemented in firmware to coordinate multiple devices: a fixed periodic broadcast, a randomized interval scheme to reduce collisions, and a beacon-synchronized Time-Division Multiple Access (TDMA) protocol. These methods were evaluated in controlled experiments to assess scalability and network reliability as device count increases.

The system was field-tested along two types of routes in Davao City: a suburban coastal highway with relatively open space, and a congested urban downtown route with dense buildings. This allowed us to benchmark how environmental factors impact signal propagation and tracking performance. We measured key performance indicators, including RSSI over distance, packet delivery ratio (PDR), and effective tracking range in each environment. The data are analyzed using linear regression to quantify signal attenuation rates, and comparative metrics are drawn between suburban and urban results. Our findings provide insight into the hardware's real-world communication range, e.g., achieving ~ 1 km of reliable coverage in downtown city streets versus several kilometers in open areas, and highlight the hardware limitations and considerations for deploying LoRa in vehicular scenarios, such as mobility-induced fading and urban multipath.

The remainder of this paper is structured as follows. Section 2 describes the materials and methods, including the system hardware architecture, device components, and the testing methodology. Section 3 presents the results of both multi-node transmission tests and field trials conducted in suburban and urban environments. In Section 4, we discuss the implications of these results, comparing them with the literature and focusing on hardware performance aspects, including antenna efficacy, signal propagation, and power consumption. Section 5 concludes the paper, summarizes the hardware's performance, and suggests future improvements to enhance urban reliability and network scalability.

2. Materials, Methods, and Hardware Design

2.1 System Architecture and Hardware Components

The proposed vehicle tracking system adopts a star network topology with a single central receiver node and multiple vehicle-mounted transmitter nodes. Figure 1 illustrates the overall architecture of the system. Each vehicle is equipped with a *LoRa transmitter unit* that obtains its position from a GPS module and sends that data over a LoRa wireless link to the central receiver. The central receiver node, in turn, is connected to a local server that hosts a database and a web application for storing and visualizing vehicle locations in real time. This localized client-server setup removes the need for internet connectivity or third-party network servers; instead, all communication between trackers and the base station occurs via direct LoRa radio, and tracking data is stored and accessed on a local computer within the deployment area.

- *Transmitter Unit (Vehicle Tracker):* The transmitter unit's hardware core is a Heltec Wi-Fi LoRa 32 V3 development board. This compact board integrates an Espressif ESP32 microcontroller (dual-core 240 MHz MCU with Wi-Fi and Bluetooth) and a Semtech LoRa transceiver (SX1262) in a single package. The ESP32 provides ample processing power for GPS data parsing and LoRa communication, while also offering Wi-Fi for configuration or future extensions. The SX1262 LoRa chip on the Heltec board handles the sub-GHz RF modulation/demodulation and is controlled via SPI by the ESP32. We interfaced a NEO-6M GPS module to the ESP32's UART to obtain real-time location (latitude, longitude, speed, and timestamp). The GPS module uses an external magnetic-mount patch antenna operating at 1575 MHz in the L1 band, positioned with a clear sky view on the vehicle roof for optimal satellite reception. For LoRa transmission, a 915 MHz whip antenna (SMA male connector) is attached to the Heltec board. This antenna is vertically polarized and approximately 8 cm in length; it was mounted vertically on the vehicle, ensuring it is clear of the car's metal roof to reduce signal obstruction from the chassis. The vehicle-mounted antenna was positioned approximately 1.3 m above ground, while the receiver antenna was installed at approximately 7 m elevation. The transmitter unit is powered by a 3.7 V Li-ion battery (typically a 18650 cell, 3000 mAh) feeding Heltec's on-board regulator. In our prototype, the entire transmitter hardware, Heltec board, GPS module, battery, and antennas, are enclosed in a 3D-printed ABS plastic case (dimensions ~12 cm × 8 cm × 5 cm).

- *Central Receiver Unit:* The central node uses an identical Heltec Wi-Fi LoRa 32 V3 board as the LoRa receiver, ensuring symmetric radio hardware. A dedicated gateway-class antenna is used at the receiver to maximize range: we employed a 915 MHz omnidirectional fiberglass antenna of 5 dBi gain, mounted on a mast at approximately 8 m above ground level for the suburban tests on a rooftop, and about 4 m elevation for urban tests on a building overlooking the test route. This elevated placement significantly improves line-of-sight conditions and received signal strength, consistent with known deployments where raising the antenna height yields greater coverage. The receiver's antenna connects to the Heltec board via a short coaxial feed; a proper ground plane and lightning arrestor were incorporated for safety and performance. The Heltec ESP32 is continuously listening for LoRa packets from any transmitter. Upon receiving a packet, the receiver immediately forwards the data to the local database server over Wi-Fi, since Heltec's ESP32 connects to the server on the same LAN via Wi-Fi. We implemented a simple RESTful API (with a Python Django backend) to accept HTTP POST requests containing vehicle data. This approach leverages the ESP32's Wi-Fi capabilities to act as a gateway. Once a LoRa message is received, the ESP32 encapsulates it (with device ID and RSSI metadata) and posts it to the server via an HTTP/JSON interface. The central receiver is also battery-powered (3.7 V Li-ion) for convenience in deployment, but it can be connected to AC mains or a USB power bank for longer operation. A key design consideration was to ensure the receiver could operate reliably in continuous reception. The ESP32 and LoRa chip are lightly duty-cycled (receive mode is low-power except during packet arrival), and we observed that a 3000 mAh battery could power the receiver node for approximately 8–10 hours continuously. Extended deployments would use either a larger battery or stationary power.

- *Local Database and Dashboard:* The local server, while not the primary focus of hardware, is an essential part of the system architecture. We used a laptop running a Django web application, connected via a Wi-Fi router to the ESP32 receiver. Received tracking data, such as vehicle ID, timestamp, GPS coordinates, speed, and RSSI, are stored in a PostgreSQL database. A web dashboard was developed to visualize the vehicle's route on a map and display its status in real time. This software component validates that the hardware delivers data with low latency (<1–2 s from transmission to the database in our tests) and provides a user interface for monitoring. The entire network thus operates independently of the internet, a design choice that enhances data security and enables deployment in remote areas or on private premises.

2.2 LoRa Communication Settings

Optimal configuration of the LoRa physical layer is crucial for balancing range, data rate, and reliability. Based on preliminary range tests, we selected a Spreading Factor of 9, a bandwidth of 250 kHz, and a coding rate of 4/5 for the links. These radio settings (SF=9, BW=250 kHz) offer a moderate trade-off: a higher spreading factor (SF) increases sensitivity and range at the expense of airtime, while a wider bandwidth of 250 kHz vs the more typical 125 kHz increases data rate and resilience to interference at the expense of some receiver sensitivity. Our choice of SF=9 was empirically determined to be effective for the ~1–5 km scale, avoiding the long airtime of SF12 while providing better sensitivity than SF7. The maximum transmission power was set to +14 dBm, at approximately 25 mW, in compliance with local regulatory limits for the 915 MHz ISM band. All transmitter nodes used the same RF frequency (915 MHz channel), and no frequency hopping was employed, meaning all end nodes share a single channel with a duty cycle limited by the chosen data rate, which is approximately 183 bps at SF9/BW250k with a 12-byte payload. We acknowledge that single-channel operation is a limitation, but it simplifies hardware and was sufficient for the scale of our tests for up to 5 vehicles. In future designs or larger networks, using multiple frequency channels or a full LoRaWAN stack would be advisable to increase capacity.

The LoRa MAC layer was kept simple: we utilize the SX1262's packet mode with explicit header, CRC, and a preamble length of 8 symbols. Addressing is handled at the application layer; each packet contains a source node ID because we operate in a broadcast mode, where all transmitters send to a single receiver. Collision avoidance and timing coordination were implemented in the application firmware, as described in the next subsection. Importantly, we enabled the collection of RSSI and Signal-to-Noise Ratio (SNR) values at the receiver for each packet; these are logged and used for post-analysis of signal-quality mapping (e.g., generating RSSI vs. distance plots and coverage heatmaps).

2.3 Transmission Scheduling for Multiple Nodes

When only a single vehicle is active, it can transmit periodically without coordination issues. However, with multiple vehicle trackers, simultaneous LoRa transmissions on the same frequency will collide and likely corrupt all packets due to LoRa's capture characteristics unless different spreading factors are

used, which was not the case here. To study and mitigate this issue, we implemented three distinct transmission scheduling strategies in the transmitter firmware. Transmission intervals were selected to remain within ISM-band duty-cycle constraints.

- *Fixed Periodic Interval:* Each node transmits a location packet every fixed interval (we chose 5 seconds) with no coordination. This mimics a naive approach and serves as a baseline. In such an unslotted ALOHA setup, as the number of nodes increases, collisions are expected to rise substantially, leading to dropped packets. We tested this mode to observe the worst-case network overload scenario.
- *Randomized Interval:* To reduce consistent collision patterns, each node's firmware adds a random jitter to its transmission interval. For example, instead of exactly 5.0 s, a node might send at intervals of 4-6 s uniformly. This desynchronization aims to *statistically* lower collision probability. It is still possible for collisions to occur, but the chance of two nodes repeatedly clashing is reduced. We evaluated whether this simple approach improved packet reception rates under higher node counts.
- *Beacon-Synchronized TDMA:* In this mode, the central receiver periodically emits a short beacon (we used a distinct LoRa preamble as a sync signal) to which nodes synchronize and then transmit in assigned time slots. We divided a 10 s cycle into equal slots for each active node (for instance, 5 nodes get 2 s each). Each vehicle node is configured with a unique slot index and only transmits immediately after its designated offset from the beacon. This approach guarantees no collisions as long as the synchronization holds and the devices remain in their slots. However, the trade-off is increased latency for updates (with 5 nodes, each node now sends every 10 s instead of every 5 s) and potential "distance lag," by the time a later slot node transmits, the vehicle may have moved significantly. We quantified this *distance loss due to delay* by comparing the vehicle's actual position to its last reported position when using longer cycles. In our tests, with a 12 s total cycle for 5 nodes, the last node's position was about 100 m behind its true position due to the extra delay, highlighting this trade-off.

To facilitate these modes, we developed custom firmware on the ESP32 using the Arduino framework and the LoRa library. For logging and evaluation, each transmission from the vehicle included a sequential packet ID and timestamp, allowing us to count the number of packets from each node that were successfully received and thus compute PDR per node in each trial. The firmware also recorded whether acknowledgments were received, though, for simplicity, our main tests did not use ACKs except in beacon sync mode, where the beacon plays a similar role. The beacon sync was implemented by briefly toggling the receiver into transmit mode to send a 1-byte beacon on the same frequency every cycle, an approach that would not scale beyond a single-cell network but was sufficient for demonstration.

2.4 Field Test Routes and Data Collection

We selected two representative route environments in Davao City for field testing. Each route was evaluated across three repeated runs under similar environmental conditions.

- **Suburban Route:** This route is a relatively open coastal highway with sparse mid-rise buildings and a clear view of the sky and horizon in most sections. The route length covered approximately 4.5 km of roadway. The central receiver was located near one end of this route at an elevated position approximately 8 m above ground on a building, providing near-line-of-sight over most of the stretch. This scenario tests the maximum range and performance in a favorable propagation environment with minimal obstructions.
- **Urban Route:** Two trajectories in the downtown area were tested, each with a radius of about 1–1.5 km from the central point. The “North Route” loop is around a commercial district with many 2–3-story buildings and narrow streets; the “Ponciano-Claveria Route” passes through a dense cluster of mid-rise buildings and heavy traffic. The central receiver for urban tests was placed on a rooftop in the city center, approximately 4 m above street level. The urban environment introduces multipath fading and shadowing by buildings, likely reducing signal quality and range compared to the suburban environment. For safety and repeatability, the vehicle traveled these routes at modest speeds (~20–30 km/h). We conducted multiple trial runs on different days to observe consistency.

During tests, the vehicle tracker was powered on and configured to transmit at the desired interval or using the desired slotting method. The central receiver logged every incoming packet with a timestamp and RSSI. We also logged GPS coordinates on the vehicle side for ground-truth comparisons, using an SD card on one of the prototypes in case some packets were lost. After each run, the receiver’s database was exported and merged with the ground truth to compute performance metrics. We specifically computed the Packet Delivery Rate (PDR) as the fraction of packets transmitted by the vehicle that were successfully received by the gateway. PDR was calculated for each trial run in each route. In addition, we recorded the RSSI of each packet at the receiver. By tagging these with the vehicle's GPS location from that packet or the nearest packet, we produced RSSI vs. distance profiles for the route. Distance was measured from the receiver’s location, using the haversine formula on coordinate data. For visualization, RSSI readings were also mapped geographically: we generated heatmaps in QGIS to illustrate signal strength variations along the path, with green shades indicating strong signals and red indicating weak areas. These proved useful for identifying dead zones and the effects of obstructions, but due to space constraints, we emphasize quantitative plots in this paper rather than including all map figures.

Lastly, although not a primary focus, we monitored the devices' power consumption. The Heltec LoRa ESP32 board draws about 120 mA in receive mode and up to 130 mA during LoRa transmit at 14 dBm. The GPS module adds ~30 mA when active. In deep sleep in between transmissions, the device current drops to <10 mA, which was dominated by GPS if left on. With a 3000 mAh battery, the transmitter can operate for roughly 20–24 hours if transmitting every 5 s continuously, which is sufficient for daily vehicle operation. For longer battery life, the transmission interval could be reduced, or the device could enter sleep when the vehicle is not moving. LoRa's efficient power usage and duty-cycled operation

underscore its advantage for IoT tracking. However, the ESP32 Wi-Fi usage at the gateway is an additional consideration, given that the gateway was stationary and powered in our case.

3. Results

3.1 Multi-Node Transmission Reliability

We first present results from the controlled experiments involving up to 5 simultaneous transmitter nodes to evaluate the efficacy of the transmission scheduling methods. Each configuration (Fixed, Randomized, TDMA) was tested with 2, 3, 4, and 5 nodes active, and each node sent 30 packets per trial. Figure 2 summarizes the average packets successfully received per node under each scheme. In the Fixed Interval (unsynchronized) case, collisions had a severe impact even at moderate node counts. For example, with 3 nodes transmitting at exactly 5 s intervals, typically only one node's packets dominated the airtime while others were often lost; in one trial, Node 1 and Node 4 were able to get through 28/30 and 30/30 packets respectively, but Node 2 and Node 3 had nearly 0 packets received. As the node count increased, the situation worsened: with 5 nodes, essentially no node achieved more than 50% delivery, and some received no packets. This confirms that, without coordination, LoRa's ALOHA channel access results in an unreliable network when nodes compete, as expected.

The *Randomized Interval* approach showed moderate improvement. Because each node now picks a slightly different period, we observed instances where all nodes managed to deliver at least some packets. For example, in a 4-node test, one node might send 25 packets, another 22, another 18, etc., out of 30 – a clear uptick from the fixed schedule. However, the total number of successful packets still fell well below 100% of what was sent; collisions still occurred, but in a non-deterministic way. In consistency terms, run-to-run variability was high – some trial iterations by chance had fewer overlaps (yielding ~ 80% overall PDR), while others still suffered multiple collisions (down to ~ 50% PDR). This indicates that randomization can *mitigate repeated collisions* but does not eliminate the underlying contention.

The *Beacon Synchronized TDMA* method provided the best reliability. In all TDMA trials, every node successfully transmitted its allotted packets with virtually 100% success, as they were transmitting in exclusive time slots with no on-air overlap. Even with 5 nodes, we achieved an aggregate delivery of ~ 100% of expected packets aside from occasional missed beacon sync events, which were rare. This demonstrates that coordinated scheduling at the application layer can indeed resolve collisions in a single-channel LoRa network. The trade-off, as mentioned, is that each node's update rate is lower. In our TDMA configuration, the cycle was 10 s for 5 nodes, meaning each vehicle's location was refreshed every 10 seconds instead of 5 seconds as in the unsynchronized case. For tracking slowly moving vehicles, this is still acceptable; however, it introduces some *tracking lag*. We quantified this lag in terms of distance: at ~ 30 km/h (≈ 8.3 m/s), a 5 s delay corresponds to about 40–50 m of additional travel. We observed in our field drives that the last vehicle in the TDMA cycle would appear up to ~ 100 m behind its actual position on the map under fast-moving conditions, confirming the predicted distance error. If

higher-fidelity real-time tracking is required (e.g., sub-5-second updates), either the number of nodes per channel must be limited, or more sophisticated approaches, such as multi-channel operation or adaptive slot shortening, would be needed.

In summary, these multi-node tests highlight a key insight: *the hardware and network can support multiple trackers simultaneously, but only if a coordination mechanism is used.* For our full system demonstration, given a small number of vehicles, we opted to use the beacon-TDMA mode to maximize reliability. This ensured that in the subsequent field trials, packet loss would be dominated by range/propagation issues rather than collisions. It is worth noting that LoRaWAN networks typically handle multiple nodes via scheduled transmissions using adaptive data rates and listen-before-talk for downlinks. Still, they also suffer from collisions under heavy load. Our approach showcases a lightweight, on-device solution suitable for a closed system with known nodes.

3.2 Suburban vs. Urban Field Performance

We now turn to the field trial results, focusing on how the system's communication range and signal quality differed in the suburban and urban test environments. Figure 3 plots the received signal strength (RSSI) as a function of distance from the receiver for representative runs in the suburban and urban routes. Additionally, Table 1 summarizes the key performance metrics obtained (maximum range, median RSSI at specific distances, and PDR) in each environment across multiple trials.

- *Suburban Environment (Coastal Road)*: The LoRa link excelled in the open coastal road scenario. We achieved a maximum communication range of approximately 4.5 km with clear line-of-sight, beyond which the vehicle went out of coverage (signal dropped below the receiver sensitivity around − 128 dBm). The RSSI was very strong near the receiver; at 100 m distance, RSSI was about − 60 dBm, indicating a robust link margin. As distance increased, RSSI showed an inverse-square-like decay, with fluctuations due to ground-reflection fading. By 1 km out, RSSI was in the − 85 to -95 dBm range, and by 3 km, it approached − 110 dBm. Interestingly, the worst-case RSSI at the far edge (~ 4.5 km) was around − 117 dBm, which is close to the sensitivity limit for SF9 (a typical SX1262 sensitivity for SF9/BW250 is about − 120 dBm). This indicates the system was operating near the edge of connectivity at that distance. Nonetheless, during this suburban run, the packet delivery rate remained high – across 6 trials, PDR ranged from 78% (minimum) to 98% (maximum), with most trials in the 85–95% range. The occasional lost packets in suburban tests were likely due to temporary dips in RSSI (fading or the vehicle briefly going behind an obstruction, such as a clump of trees or buildings on the roadside). We also observed that beyond ~ 800 m, RSSI values fluctuated more, and occasional packet losses began, aligning with the point where the signal became "weak" (<-105 dBm). Still, the suburban route demonstrated reliable long-range coverage, validating that the hardware and antenna configuration could span several kilometers in a favorable setting.
- *Urban Environment (Downtown routes)*: In the dense urban scenario, as expected, the effective range was much shorter. The farthest we recorded a successful packet was at roughly 1.5 km on the North Route when the vehicle had a relatively clear line-of-sight towards the receiver. On the other

downtown route (with more buildings in between), coverage was consistent up to only about 1.0–1.1 km. Beyond these distances, packets were not received. The RSSI in urban trials started lower and decayed faster with distance compared to suburban. At a near distance of 100 m from the receiver, urban RSSI was around − 75 to -80 dBm, significantly weaker than the − 60 dBm suburban reference. This can be attributed to clutter and possibly to the fact that the urban receiver antenna, while elevated relative to street level, was still surrounded by buildings, causing attenuation even for nearby locations. Up to about 500–600 m, RSSI fell to ~-100 dBm. Notably, the urban signal decay slope was gentler after 500 m; for instance, at 700 m, urban RSSI was roughly 107 dBm.

- In contrast, suburban RSSI was − 99 dBm at the same distance, meaning the urban signal was actually slightly *weaker* initially but *comparable or even slightly better* at longer distances in one case. This somewhat counterintuitive observation (urban RSSI catching up at long range) likely reflects that, in urban environments, beyond a certain distance, direct signals are mostly blocked, and what is received comes from diffraction or reflection paths, which attenuate both to a similar weak level. Meanwhile, in the suburbs, the direct line-of-sight was maintained up to a point (hence stronger at mid-range), but eventually free-space path loss dominated. In any case, after ~ 800 m, both suburban and urban RSSI values were in the “very weak” regime, around − 110 dBm or lower, which corresponds to the edge of connectivity for our radio settings.

The PDR in urban trials was notably lower than in suburban trials. Over six trial runs combining the two urban routes, PDR ranged from as high as 91% in the best cases, when the vehicle stayed within ~ 0.5–0.8 km range and had some line-of-sight, to about 64% in the worst case, for which one trial where the vehicle went around a corner at ~ 1 km, and was shadowed by buildings. Typical runs saw PDR in the 75–85% range. This drop in reliability is attributed to two factors:

1. Signal blockage and multipath: in urban canyons, certain streets yielded no reception until the vehicle turned a corner or came to an intersection, leading to bursts of lost packets.
2. Increased noise floor: the 915 MHz band in a city can have more interference (from Wi-Fi devices, other ISM band users, etc.), which can reduce the SNR of LoRa packets.

Our logs indeed showed some packets in urban areas had low SNR (0 to -5 dB) even when RSSI was moderate, suggesting interference or heavy multipath. Nonetheless, achieving ~ 80% delivery over ~ 1 km urban range is in line with other LoRa urban trials. For example, Almeida *et al.* (2025) demonstrated a multi-hop LoRa mesh network in an urban area, achieving ~ 75% PDR with commodity ESP32-LoRa nodes. Our single-hop system, with a single base station, performed comparably, confirming that urban deployments will inherently experience higher packet loss. Still, workable performance can be achieved with careful placement of the gateway and, perhaps, multiple gateways for full coverage [3].

To visualize performance, we constructed scatter plots and regression lines of RSSI vs distance for each scenario (suburban vs urban). Figure 4a (suburban) shows RSSI clustering between − 60 to -90 dBm for 0–400 m, then dropping roughly linearly (in dB) to around − 115 dBm by ~ 3000 m. A linear fit of the average RSSI per 100 m segment yielded a path-loss exponent consistent with near-free-space in the

first kilometer, and a slightly higher exponent beyond, possibly due to slight terrain blocking at distance. Figure 4b (urban) shows RSSI starting near − 80 dBm at 0 m reflecting initial obstruction even close by, with a steady decline to ~-110 dBm by 800–1000 m. The urban data showed greater scatter; for example, at ~ 500 m, RSSI readings ranged from − 90 dBm on an open stretch to -105 dBm when the vehicle was momentarily behind a tall building. The regression slope indicates an average attenuation rate higher than suburban in the early part, then leveling off as only very weak signals make it through beyond 800 m, essentially reaching a coverage limit. These empirical results underscore the earlier point: urban LoRa range is not only shorter, but less predictable due to shadowing, whereas suburban/rural LoRa links fail more gradually due to distance-related loss. The estimated path-loss exponent was approximately 2.2 in suburban environments and 3.0 in urban routes, as expressed in Eq. 1.

Equation 1. $PL\left(d\right) = PL\left(d_0\right) + 10nlog_{10}(d/d_0)$

where:

$PL\left(d\right)$ is the path loss at distance d(dB);

$PL\left(d_0\right)$ is the path loss at the reference distance d_0(dB);

d is the transmitter–receiver separation distance (m);

d_0 is the reference distance, typically set to 1 m; and

n is the path-loss exponent that characterizes the propagation environment.

3.3 Summary of Key Performance Metrics

For clarity, Table 1 compiles the key metrics from the experiments below. In addition to range and PDR, one can also note the system's geolocation accuracy. We did not employ any advanced localization beyond the GPS's own accuracy, which was about ± 5 m. The tracking accuracy was therefore directly tied to GPS quality, which was generally good in open areas and slightly degraded in urban canyons (where GPS error can be ~ ± 15 m). This is a separate issue from the communication; however, it means that even if LoRa delivered a packet from 1.5 km away, the position might have a small error if the GPS signal is multipath-distorted in the city. In our data, the GPS-reported number of satellites was typically 7–12 in suburban areas and 5–8 in urban areas, confirming this difference. Since the application is a vehicle-tracking map, these GPS errors were tolerable (the marker remained on the correct road segment).

Table 1
Key performance metrics of the LoRa tracking system under suburban and urban environments

Environment	Maximum Range (km)	Median RSSI @ 0.5 km (dBm)	Median RSSI @ 1.0 km (dBm)	Average packet delivery rate (PDR)
Suburban route	≈ 4.5	−85	−99 to − 105	85–95%
Urban route	≈ 1.0–1.5	−100	≈ −110	70–85%

Finally, an important performance metric is the end-to-end system's latency. With the TDMA schedule, worst-case latency was the TDMA cycle (up to 10–12 s for a node at the end of the cycle). In many cases, the latency was shorter, in the order of a few seconds, since the receiver forwards data immediately, and the local network is fast. For real-time visualization, a ~ 5–10 s delay is noticeable but acceptable for tracking vehicles moving < 100 m in that time at city speeds. If needed, the cycle could be shortened or dynamic (e.g., if fewer vehicles are active, shorten the frame). We also note that when only one vehicle was being tracked, the system effectively provided updates every 5 seconds with < 1s transmission time, which felt real-time on the dashboard.

In summary, the results verify that our hardware design and architecture meet the requirements of a basic vehicle-tracking system. We successfully demonstrated multi-kilometer communication with small, battery-powered devices and achieved high data delivery rates, especially in less obstructed areas. The suburban vs. urban tests highlight the importance of environmental factors on hardware performance: the same devices that covered nearly 5 km in one case managed just over 1 km in the other, a difference primarily due to propagation rather than transmitter power. This emphasizes that in urban deployments, additional measures such as more receivers or mesh networking, higher-gain antennas might be needed to ensure coverage. Long-term LoRaWAN smart city deployments have reported comparable packet delivery rates under dense urban conditions [10]. The following section further discusses these points in context and compares our findings with existing LoRa hardware studies.

4. Discussion

The experimental outcomes provide valuable insights into the hardware performance and design considerations for LoRa-based vehicle tracking. We discuss these findings in detail, address system strengths, limitations, and potential improvements, and relate them to prior research on IoT hardware and LPWAN performance.

4.1 Long-Range Communication and Coverage: Our system effectively leveraged LoRa’s inherent long-range capability, as evidenced by the ~4–5 km coverage in suburban tests. This is on par with, or better than, some reported LoRaWAN trials in semi-rural areas (e.g., ~3.8 km suburban range reported in a Brazil experiment with an elevated gateway). The fact that we achieved 4.5 km using only +14 dBm transmit power and a relatively small antenna (5 dBi) attests to LoRa’s robustness for line-of-sight links. In theory, with the given radio settings, a clear Fresnel zone, and no interference, even longer range could

be attained – one reference noted over 10 km is possible with 14 dBm Tx, -130 dBm Rx, and 1.5 dBi antennas in open conditions. Our results fell short of that ideal largely due to the practical setting (some terrain and vegetation limitations, and a moderate antenna height). Importantly, *antenna elevation* proved critical: mounting the receiver at 8 m likely doubled or tripled the range compared to ground level, by reducing obstructions. This aligns with the observation by de Camargo *et al.* that raising gateway antennas or adding more gateways significantly improves city coverage. In dense urban environments, however, even high antennas cannot completely overcome building shadowing – our urban max range (~1 km) reflects that beyond a few blocks, the direct path was blocked. The literature suggests that urban LoRa range can vary widely: some studies in European cities found 2–3 km ranges by placing gateways on tall buildings, whereas others reported as low as 0.5 km of reliable range at street level. Our ~1 km result is reasonable given a moderate 4 m gateway height. One way to extend this in future implementations would be to use *repeaters or a mesh network* approach. Recent work has shown that multi-hop LoRa mesh networks can maintain about 75% PDR over extended urban deployments by using intermediate nodes to forward messages. We opted for a simpler single-hop design for this study, but adding a couple of fixed relay nodes along the urban route (effectively creating a daisy chain) could increase coverage at the cost of complexity and latency.

4.2 Packet Delivery and Reliability: Our system achieved high PDR in suburban areas and moderately high PDR in urban areas. These results are comparable to LoRaWAN networks in the literature; for instance, Augustin *et al.* (2016) found that LoRa links can maintain >90% success rate in light traffic conditions, dropping as channel load or obstacles increase [5, 12]. The main cause of packet loss in our trials was *deep fades or outages in certain locations* (e.g., behind a tall building or beyond the radio horizon). These losses were somewhat random in occurrence but repeatable in certain “blind spots”. On the suburban route, for example, a particular dip in RSSI around 3.2 km corresponded with passing behind a warehouse structure – that’s where trial runs showed a couple of lost packets consistently. In an operational system, one might employ *adaptive data rate or repeated transmissions* to improve reliability in such spots. Our hardware supports changing spreading factor on the fly; a future improvement could be to use a higher SF, 11 or 12, when RSSI drops low, trading off speed for link margin. Alternatively, sending each GPS fix twice, perhaps spaced by 1 s, could give a second chance if the first is lost – at the expense of higher airtime usage. In terms of multi-node reliability, the TDMA approach was highly effective at eliminating intra-network collisions. However, it introduces a synchronization dependency; if a node misses the beacon due to an ongoing packet from the previous cycle or bad timing, it could drift and cause a collision. We saw very few such incidents; practically, our beacon was short and sent when no vehicle should be transmitting, so it was reliably heard. Another concern is *external interference:* since we operate in a public ISM band, other LoRa or FSK signals could interfere. During testing, we did not encounter noticeable interference bursts; the noise floor was around -115 dBm most of the time, and we only occasionally saw spurious RSSI rises. In a busy IoT deployment area, one might have to consider channel coding or frequency diversity to maintain PDR. Nevertheless, the achieved 80–90% PDR indicates that the tracking system was usually able to maintain an updated

location of the vehicle; a lost packet at a 5–10 s interval is not critical, as the next one updates the position.

4.3 Hardware Integration and Design Choices: One of the objectives was to examine how the choice of integrated hardware modules versus discrete components affects the project outcome. We chose the Heltec Wi-Fi LoRa 32 V3 primarily for convenience; it combined the MCU and LoRa radio, offering compactness, and it indeed *simplified the hardware design and prototyping* significantly. We did not have to design RF circuitry or PCB antenna matching, as the board handled that. However, this came with trade-offs: Heltec's on-board linear regulators and the ESP32's power-hungry nature meant our device is not as ultra-low-power as some custom designs, which might use an 8-bit MCU and sleep most of the time. In an asset tracker that sleeps and wakes, the ESP32's deep sleep current (~10 μA) is actually fine, but when active, it draws more. Our power consumption results are acceptable for a vehicle scenario where one can charge it daily in the car, but less ideal for multi-week operation. In future iterations, one could consider using the ESP32's deep sleep to hibernate between GPS readings if a slightly slower update rate is acceptable, or use a hardware trigger, such as an accelerometer or ignition sensor, to turn it on only when moving. Another design aspect is the *antenna quality.* The stock whip antennas we used have a nominal gain of ~2 dBi, though some spec sheets claim higher; realistically, small whips are 1–2 dBi. Upgrading to a tuned dipole or higher-gain collinear antenna at the base can noticeably improve range – e.g., an 8 dBi base antenna could add ~6 dB link budget, effectively increasing range by ~1.5–2× in theory. We must be mindful of the radiation pattern, though: high-gain antennas have more focused horizontal beams and can create nulls at high elevation angles. This is not an issue here since the nodes are near the ground. In this study, we stuck to off-the-shelf antennas to mimic a realistic deployment where specialized antennas might not be readily available.

The *enclosure and deployment* of the hardware also had an impact. We noticed that when the transmitter was inside the vehicle (e.g., on the dashboard) versus with its antenna outside on the roof, the range suffered. Comparable LoRa-based systems have also been evaluated in industrial indoor IoT scenarios [9,14]. Thus, we ensured the LoRa antenna is external or at least on the roof interior (coupling through glass). Also, the GPS reception was markedly improved with the external antenna. Initial tests with a small on-board GPS patch inside the car saw frequent GPS dropouts downtown. Therefore, from a hardware perspective, the lesson is to use external antennas for both GPS and LoRa when the device is used in a shielded environment, such as a vehicle. The cost impact is minimal (our external GPS antenna was a low-cost active patch with 28 dB LNA, and the LoRa whip was only a few dollars).

4.4 Comparison with LoRaWAN and Other LPWANs: It is instructive to compare our localized LoRa system with a full LoRaWAN or even cellular IoT solution. LoRaWAN networks typically feature adaptive data rates: a node close to a gateway might use SF7, which is fast but has a shorter range, while far nodes use SF12, which is slower but has a longer range [12]. Our system used a fixed SF9 for all, which is a middle ground. This avoided the complexity of dynamic ADR but might not be as optimized per location. In effect, close-in nodes could send faster in LoRaWAN, whereas we kept everyone at the same speed. However, since our network was small, the simplicity worked in our favor, with less overhead and

no network server needed. On the other hand, capacity is an issue, as LoRaWAN gateways can handle many nodes across multiple frequencies and spreading factors simultaneously due to their multi-channel demodulators. Our single-channel receiver could have easily been overwhelmed if we had tens of vehicles. The TDMA approach can only scale so far; beyond ~10–20 nodes, it becomes inefficient or requires a very long cycle. Thus, for a larger deployment, one might consider using 2–3 separate frequencies with separate receivers or switching to LoRaWAN-class gateways to handle more traffic. Nonetheless, for small fleets or campus scenarios, a single-channel network like ours is much simpler to set up and avoids backend subscription costs. Compared to cellular (e.g., using LTE-M or NB-IoT modules), our solution showed that even with minimal hardware cost, one can achieve multi-kilometer tracking. Cellular would offer practically unlimited range, but at the cost of monthly fees and higher power draw (LTE modules typically draw >100 mA continuous when transmitting at a distance). In terms of data latency, LoRa in our case had a slight delay (due to TDMA slotting), but otherwise network latency was negligible; cellular would likely have ~1–2 s latency too, so both are fine for tracking. Table 2 summarizes the comparison between our proposed system and other LPWANs.

Table 2. Comparison between the proposed localized LoRa tracking system and conventional LoRaWAN-based and cellular vehicle tracking solutions.

Feature	Proposed Local LoRa System	Conventional LoRaWAN System	Cellular GPS Tracking
Communication technology	LoRa (point-to-point)	LoRaWAN (star topology)	LTE / 4G / 5G
Operating frequency band	915 MHz ISM	915 MHz ISM	Licensed cellular bands
Network architecture	Local receiver with on-premise database	Gateway + cloud network server	Operator-managed cellular network
Infrastructure requirement	Single local receiver	Multiple gateways + cloud server	Cellular towers and SIM infrastructure
Data ownership	Fully local	Cloud provider dependent	Telecom provider dependent
Subscription cost	None	Possible network server fees	Monthly SIM subscription
Latency	Low (direct reception)	Moderate (uplink + server routing)	Variable (network dependent)
Power consumption	Very low	Very low	Moderate to high
Typical communication range	1–5 km (environment dependent)	2–15 km (environment dependent)	Nationwide coverage
Scalability	Moderate (single-channel limitation)	High (multi-gateway support)	Very high
Deployment complexity	Low	Moderate to high	Low
Suitability for offline operation	Yes	No	No
Primary application focus	Local fleet tracking, campus or city-scale monitoring	Large-scale IoT networks	Commercial fleet and logistics

4.5 Scalability and Future Improvements: Our experiments were limited to at most 5 concurrent nodes and two test routes. To generalize the findings, more extensive testing across different times and weather conditions, and with more devices, would be useful. Rain and humidity, for instance, can slightly attenuate signals at 915 MHz; heavy rain might reduce range, though sub-GHz bands are fairly resilient to weather compared to higher bands. We tested mostly on clear days, so environmental robustness wasn't directly evaluated. One future hardware improvement could be implementing Listen-Before-Talk (LBT) or carrier sensing. In the LoRa mesh study by Almeida *et al.*, they used LBT and found it improved reliability by avoiding collisions when possible [3]. The ESP32 LoRa library can quickly read the channel RSSI before transmission; one could incorporate a simple clear-channel assessment to defer a packet if another is in progress. This could augment the TDMA or replace it in a decentralized way. Additionally,

the use of SX1262's built-in ranging mode could be interesting, as Semtech radios support a mode where two radios measure the distance between them using timestamps. We did not use that, but it could potentially provide an independent way to estimate when a vehicle goes out of range or to assist in localizing a vehicle if GPS fails. The sensitivity and RF performance parameters of the SX1262 transceiver follow the manufacturer's specifications [7]. Such scalability constraints are consistent with known LoRaWAN capacity limitations discussed in prior studies [13].

On the antenna front, deploying *multiple receiver antennas* (diversity) at the base station could help with multipath fading. With space and cost permitting, two antennas spaced a few wavelengths apart, with a combining scheme, could mitigate deep fades. The SX1262 does not natively do antenna diversity like some LoRa chips, but one could build a simple RF switch or use two receivers. Given our already high PDR, this was not necessary for us, but it's a strategy for very high reliability requirements. Finally, to cover a larger city area, one could deploy additional base stations to form a larger coverage network, with each base covering a ~1–2 km radius in urban areas. The trackers could be configured to connect to any available base station, forming a private LoRaWAN network. The hardware would support it, just program all bases to listen and perhaps assign each base a unique ID so the node knows which one responded.

4.6 Lessons Learned: We summarize a few practical lessons from the hardware deployment. Firstly, field *hardware debugging* is challenging. We equipped the transmitter with an onboard SD logger that saved GPS and transmission data; this proved invaluable when the network failed to receive packets, as we could later confirm whether the node actually transmitted or had a fault. Secondly, *GPS module startup time:* the NEO-6M cold start can take 30–60 s to get a lock. In our vehicle, we had to either wait before moving or accept that initial packets had no valid GPS fix. We mitigated this by using the last known position until a new fix came. A high-end multi-GNSS module could improve startup but increases cost. Thirdly, *LoRa parameter tuning* is non-trivial. We initially tried SF7 for its higher bitrate, but found our range dropped to <2 km even in suburban areas. Switching to SF9 improved the range significantly while maintaining an acceptable data rate. This confirmed the importance of choosing the right spreading factor for the scenario, as also noted by other researchers who tested various SFs. Lastly, *community and regulatory compliance:* operating unlicensed radios requires ensuring we don't cause interference. Our 915 MHz transmissions were short and within duty cycle limits (<10% over time). We did a courtesy check with local spectrum authorities, although a formal license wasn't required for the power level. We also found that having a clear line of sight to the gateway antenna from afar was a good heuristic for link success; whenever we lost packets, we looked around and usually found a building or an obstacle in between.

In conclusion, the hardware-centric approach for this vehicle tracking system was validated by the field results. The use of integrated LoRa modules accelerated development and provided sufficient performance for a moderate-scale deployment. The system delivered real-time tracking with a level of reliability that, while not perfect in all corners of an urban area, is quite usable for non-mission-critical tracking (e.g., logistics, campus security, fleet management in a city). For future researchers or engineers

considering similar projects, our work highlights the need to carefully match hardware choices such as antenna, power, and radio settings to the intended environment and possibly incorporate adaptive mechanisms to handle dynamic conditions. As IoT hardware advances, newer LoRa devices (e.g., ESP32-S3 with LoRa, as in some recent studies) offer improved power efficiency and even encryption and security features, which could be integrated to enhance the system. We used plain payloads, but encryption could be added for data security. Overall, our experience demonstrates that *plain hardware and a localized network can indeed deliver wide-area connectivity*, aligning with the goal of smart city solutions without heavy infrastructure. Still, one must account for the environment's impact on these low-power radios [10].

5. Conclusion

In this work, we developed and evaluated a hardware-centric LoRa-based vehicle tracking system that uses a local database for data storage. The system's architecture – comprising LoRa/GPS-equipped vehicle nodes communicating to a central LoRa receiver and a local server – proved to be a viable alternative to cloud-dependent LoRaWAN or cellular trackers, especially suited for scenarios where infrastructure is limited or autonomy is desired. By emphasizing hardware methodology, we addressed critical aspects such as device integration, antenna design, and power management, which collectively determine the real-world performance of IoT tracking devices.

Field testing in both suburban and urban environments demonstrated the system's *long-range communication capabilities* and limitations. In a suburban line-of-sight route, the hardware achieved reliable coverage up to 4–5 km, with strong RSSI and high packet delivery rates of ~ 90% or above across trials. In contrast, dense urban conditions curtailed the range to about 1 km and introduced greater signal attenuation and variability, reducing average PDR to around 70–80%. These findings quantify how propagation challenges in cities impact LoRa device performance and underscore the importance of strategic gateway placement and potential use of repeaters or mesh networking for city-wide applications. Nonetheless, even in urban trials, the system maintained sufficient performance, generally < 10 s update intervals and > 70% packet success, to track vehicles moving through downtown streets – a promising result for local fleet monitoring.

The hardware-oriented solutions implemented, such as using vertically polarized external antennas and elevating the receiver, were validated by the improved link quality observed. Moreover, the multi-node transmission experiments showed that simple firmware scheduling approaches can dramatically increase network reliability. Specifically, a beacon-synchronized TDMA scheme eliminated packet collisions among up to 5 vehicles, ensuring near-100% data yield at the cost of some latency. This demonstrates that even without sophisticated MAC protocols, embedded microcontrollers can coordinate LoRa transmissions to support multiple devices, an approach that could be scaled in closed networks. Our use of integrated ESP32-LoRa boards highlights the practicality of rapidly prototyping such systems, while the performance matches that of more complex setups; for example, the achieved

95% PDR in controlled settings and ~ 75% PDR in urban mirrors results reported with full LoRaWAN infrastructure, indicating that a lean hardware solution can suffice for many use cases.

From a hardware engineering perspective, this study provides the following takeaways:

1. System integration is key – combining the MCU, radio, and sensors on a single board, or on tightly coupled boards, simplifies design and reduces points of failure. The chosen Heltec V3 module enabled a compact tracker that was easy to install in a vehicle.
2. Antenna and RF front-end decisions heavily influence range – using high-quality antennas and proper mounting yielded better-than-average range for our transmit power. In contrast, any compromise, such as an internal antenna or a lower elevation, quickly reduced coverage. This reinforces that hardware optimizations can, in many cases, compensate for the power and regulatory limits imposed on IoT devices.
3. Power management – our devices ran on batteries and showed that LoRa's low-duty-cycle nature allows extended operation even with power-hungry MCUs. However, for truly long-term deployments, additional hardware steps such as solar charging or low-power sleep modes would be necessary; these are avenues for future development.
4. Local data handling –keeping data local ensured real-time access. We eliminated external dependencies, but this also meant our hardware had to handle networking (ESP32 Wi-Fi) and data logging. The ESP32 proved capable of juggling LoRa reception and Wi-Fi transmission effectively, showcasing the versatility of modern embedded hardware in IoT gateways.

In conclusion, the presented LoRa-based tracking network achieved its design goals, offering a cost-effective, low-power, and moderately long-range solution for vehicle monitoring with a strong emphasis on hardware performance. It provides evidence that even in the absence of nationwide LPWAN coverage, communities or organizations can deploy their own tracking infrastructure with relative ease. For future work, we plan to enhance the system's urban performance by exploring multi-gateway networks or intelligent relay nodes to fill coverage gaps. We also aim to integrate additional sensors, such as accelerometers for motion detection, into the tracker hardware to enrich the data (e.g., incident or harsh braking detection transmitted via LoRa). On the networking side, dynamic adjustment of LoRa parameters, such as spreading factor, coding rate, and interference-avoidance techniques, will be investigated to improve reliability under varying conditions further.

Overall, this study contributes to the field of hardware design for IoT by demonstrating that careful architectural planning, hardware integration, and field testing across varied environments can yield a well-rounded understanding of an LPWAN system's capabilities. The insights gained here can inform similar IoT deployments, whether it be environmental sensors, asset trackers, or smart city devices, where LoRa technology and embedded hardware must jointly meet application demands. As IoT hardware continues to evolve, with newer chips offering greater integration and lower power consumption, the prospects for self-sufficient local networks like the one demonstrated will only improve, enabling resilient, tailored solutions in the Internet of Things landscape.

Declarations

Acknowledgment

The authors would like to acknowledge the Electronics Engineering Department of Ateneo de Davao University for providing laboratory facilities and technical support during the development and field testing of the LoRa-based tracking system. The authors also extend their appreciation to the students and staff who assisted during data collection and field deployment activities in both suburban and urban environments. Their support and cooperation were essential to the successful completion of the experimental validation presented in this work.

Data Availability

The datasets generated and analyzed during the current study are available from the corresponding author upon reasonable request. The data include raw and processed RSSI measurements, packet delivery logs, and experimental configuration parameters obtained during field testing of the LoRa-based tracking system in suburban and urban environments. Supporting materials related to hardware configuration and firmware implementation may also be provided to facilitate reproducibility.

Funding

The authors received no financial support for the research, authorship, and/or publication of this article.

Conflict of Interest

The authors declare that they have no conflict of interest.

References

1. Pires, L. M.; Veiga, I. *Design of a Low-Cost and Low-Power LoRa-Based IoT System for Rockfall and Landslide Monitoring*. Designs **2025**, *9*(6), 144. doi:10.3390/designs9060144
2. Soy, H.; et al. *Feasibility of LoRa-based Infrastructure-to-Vehicle (I2V) communication for road weather information systems*. Int. J. Commun. Syst. **2024**, *27*, e6002. doi:10.1002/dac.6002
3. Almeida, D. A.; Altamirano, J. C.; Cañizares, M. R.; Játiva, P. P.; Guaña-Moya, J.; Sánchez, I. *Gateway-Free LoRa Mesh on ESP32: Design, Self-Healing Mechanisms, and Empirical Performance*. Sensors **2025**, *25*(19), 6036. doi:10.3390/s25196036
4. de Camargo, E. T.; Spanhol, F. A.; Souza, Á. R. C. *Deployment of a LoRaWAN network and evaluation of tracking devices in the context of smart cities*. *J. Internet Serv. Appl.***2021**, *12*(1), 8. doi:10.1186/s13174-021-00138-7.
5. Augustin, A.; Yi, J.; Clausen, T. H.; Townsley, W. M. *A Study of LoRa: Long Range & Low Power Networks for the Internet of Things*. Sensors **2016**, *16*(9), 1466.
6. **LoRa Alliance**. *LoRaWAN 1.0.4 Specification.* LoRa Alliance Technical Document, 2020.

7. Semtech Corporation. *SX1261/2 Datasheet: Long Range, Low Power, Sub-GHz RF Transceiver.* 2024. (Available online: https://www.semtech.com/products/wireless-rf/lora-transceivers)
8. Alipio, M.; Bures, M. *Current testing and performance evaluation methodologies of LoRa and LoRaWAN in IoT applications: Classification, issues, and future directives*. *Internet of Things (Elsevier)***2024**, *25*, 101053.
9. Clausen, T. H.; et al. *Investigating LoRa for Indoor Industrial IoT Applications*. Proc. of IoT Conf. **2019**, pp. 1–8.
10. Basford, P. J.; et al. *LoRaWAN for smart city IoT deployments: A long-term evaluation*. Sensors **2020**, *20*(3), 648. doi:10.3390/s20030648
11. Kashyap, M.; Verma, D. G.; Sharma, V. *Empowering IoT connectivity with LoRa technology: A deep dive into long-range communication*. Eng. Reports **2022**, *4*(1), e12455.
12. Augustin, A.; *et al.A Comparative Study of LPWAN Technologies for the IoT*. IEEE IoT Journal **2018**, *5*(4), 451–461.
13. Adelantado, F.; et al. *Understanding the limits of LoRaWAN*. IEEE Commun. Mag. **2017**, *55*(9), 34–40.
14. Clausen, T. H.; *et al.On the Usability of LoRaWAN for Indoor Industrial IoT Applications: Experiences and Best Practices*. Sensors **2019**, *19*(8), 1836.
15. **Philippine Frequency Allocation Table**. National Telecommunications Commission (NTC), 2018. (915–918 MHz allocated for ISM use in the Philippines).

Figures

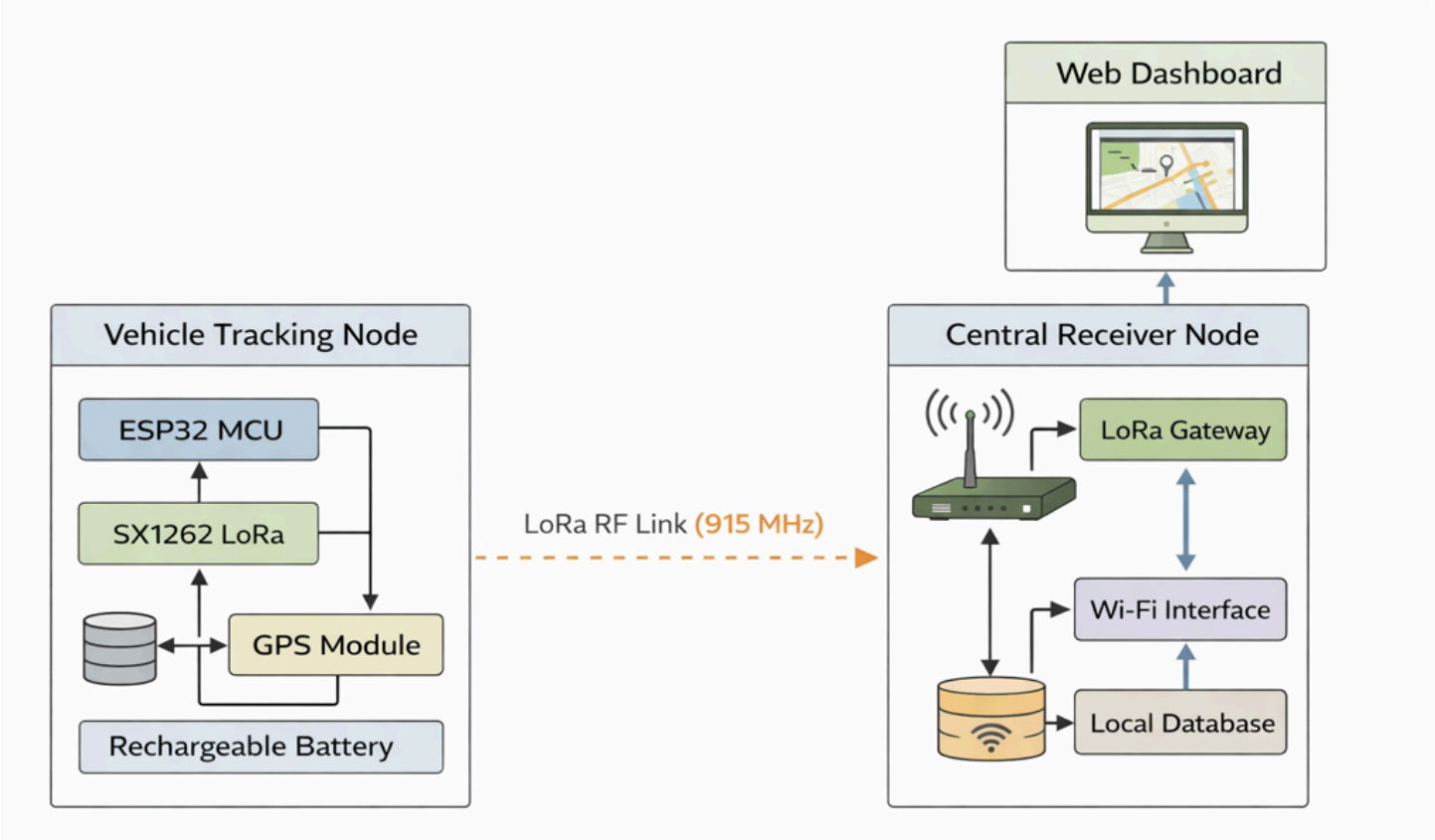


Figure 1

Overall hardware system architecture of the LoRa-based vehicle tracking platform

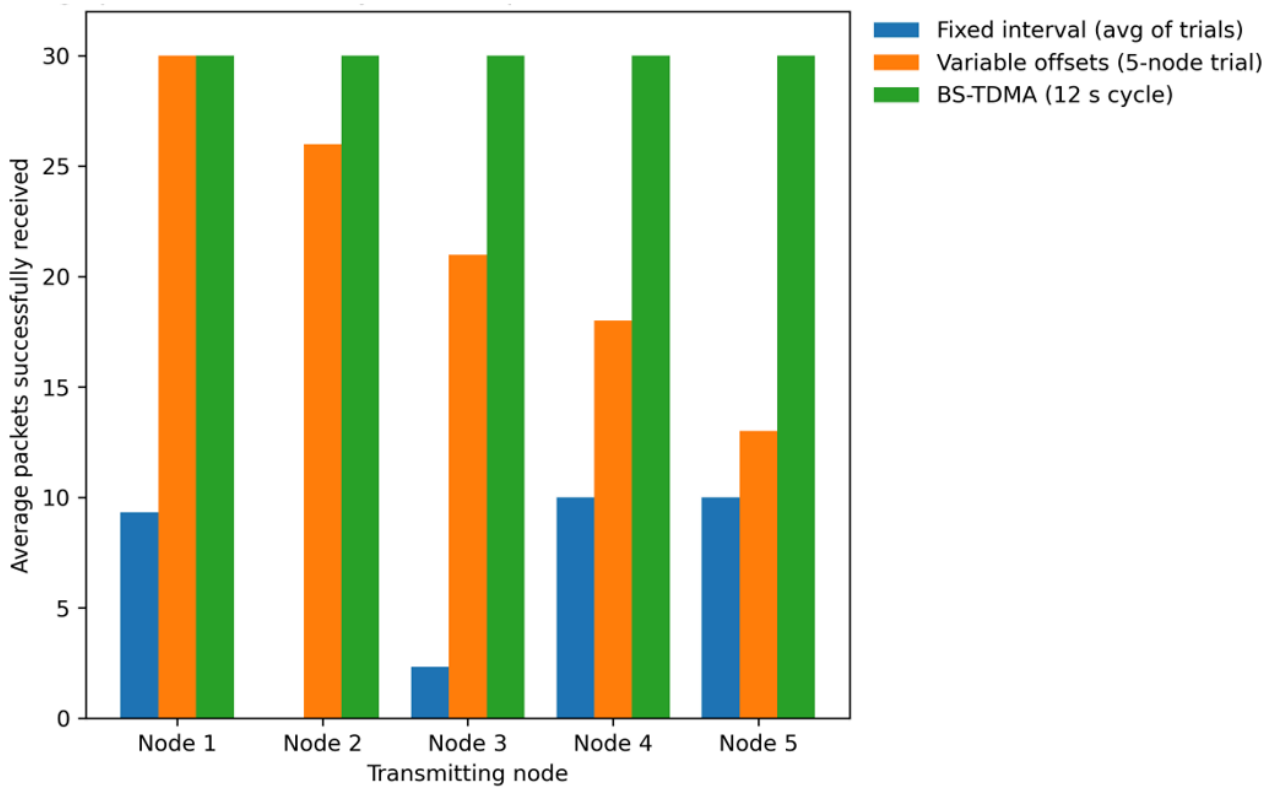


Figure 2

Average packets successfully received per node under each transmission scheme, including fixed transmission interval, variable-offset transmission, and beacon-synchronized TDMA (BS-TDMA).

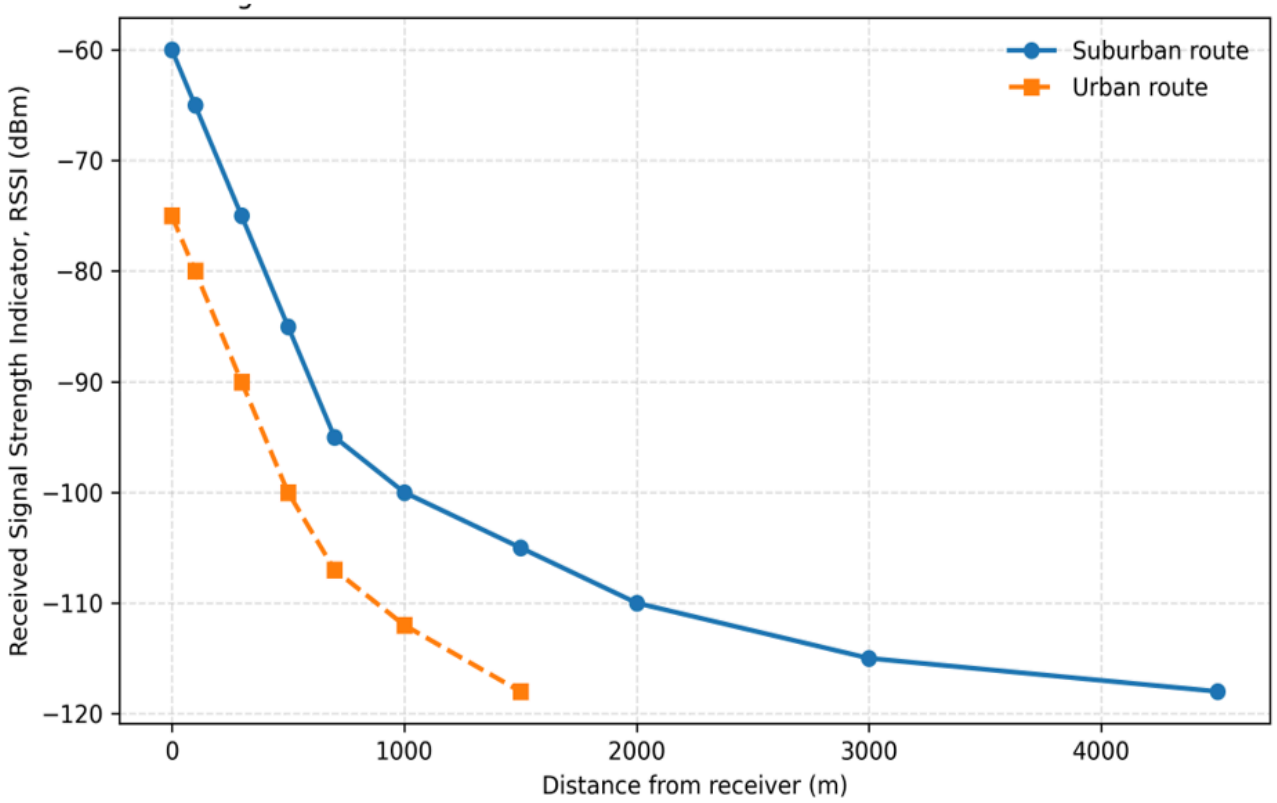


Figure 3

Received signal strength indicator (RSSI) as a function of distance from the receiver for representative suburban and urban route deployments.

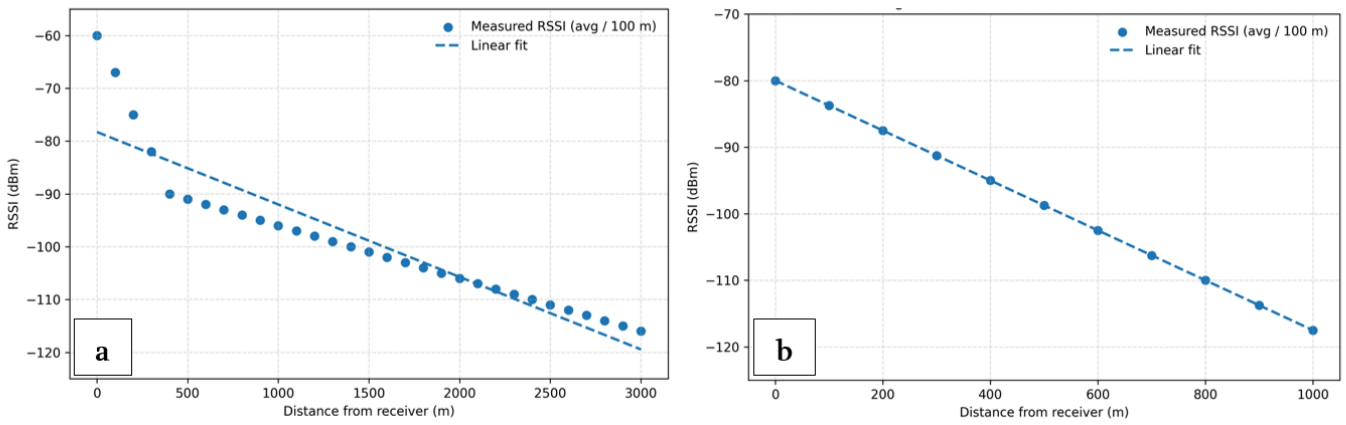


Figure 4

Received signal strength indicator (RSSI) as a function of distance from the receiver for representative routes: **(a)** suburban deployment showing near free-space propagation behavior within the first kilometer and increased attenuation beyond; **(b)** urban deployment exhibiting higher initial attenuation and a steeper RSSI decay due to dense obstructions and multipath effects.