Ad-Hoc Train-Arrival Notification System for Railway Safety in Remote Areas

Aida Eduard[a], Dnislam Urazayev[a], Aruzhan Sabyrbek[a], Daniil Orel[a], Dimitrios Zorbas[a]

[a]*Nazarbayev University, School of Engineering & Digital Sciences, Astana, Kazakhstan*

Abstract

In the last few years, a particular interest in using wireless technologies in the industrial domain in order to automate processes and increase the level of safety has been noticed. This paper introduces an affordable mobile system to notify railway workers in rural areas about the approach of trains, and thus, to enhance their safety allowing for the early evacuation of repair sites located near the rails. The system comprises three key elements: a train device, a portable station device, and wearable devices for the workers. The communication methods and the underlying protocols between these components are discussed in detail. The system has been developed for freight trains of the national railway company of Kazakhstan and has undergone extensive testing for each of its components before its final trial. The preliminary results demonstrate that the system meets the requirements in terms of evacuation time, range, and portability, while exhibiting a very low cost of manufacturing. More specifically, the system can achieve a reliable communication range of several kilometers and a maximum response time of 2.3 seconds. The cost does not exceed $500 for a set of train, station, and 5 worker devices.

Keywords: Railway safety, LoRa, ESP-NOW

1. Introduction

The transport of goods using railways constitutes an easy solution to reduce freight costs without delaying their delivery especially for long distance transports between neighboring countries [1]. However, extensive railway paths are required to be built and maintained throughout huge territories. The maintenance of railways in remote areas poses risks due to the lack of permanent infrastructures in those areas and the possible presence of extreme weather conditions [2]. EU statistics have shown that 1389 significant railway accidents took place and 683 people were found deceased, and 513 people were injured[1]. Thus, the importance of improving the safety methods for people working along the railroad about the incoming trains is rising. Sometimes, the deployment of permanent systems is not feasible because of the high cost, mainly in countries spanning large geographical areas. Kazakhstan is an example of such countries having a huge territory with very few people living between major cities. Hence, the development of permanent infrastructures is not an easy task for many reasons. The aim of this research is to propose a cost-effective, portable, and long-range notification system that can alert workers operating on or near railway tracks about the approach of trains.

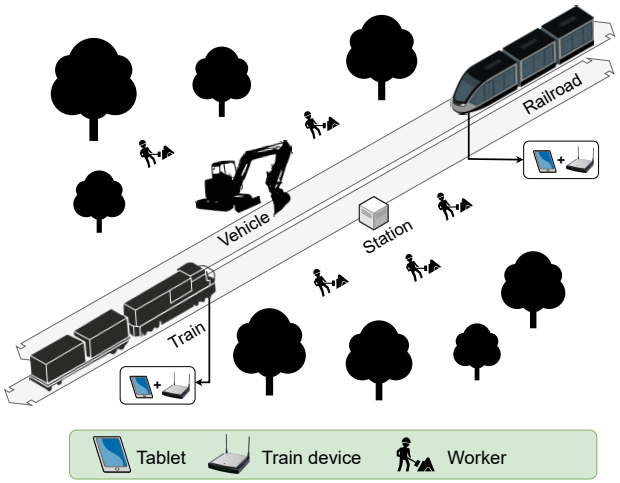


Figure 1: An example of a use-case scenario.

The system has been specifically designed for the national railway company of Kazakhstan and it is planned to be deployed in 2024 with a primary focus on freight trains that operate at average speeds not exceeding 50-60 km/h. However, modifications can be made to be adjusted to other use-cases as well. It is important to note that the proposed system is designed to complement permanent signaling systems in areas where currently there is no coverage by one of those systems. A scenario example illustrating the system's functionality is depicted in Figure 1. In this scenario, workers need to evacuate the repair site before a train arrives. The evacuation process may take several seconds as machinery and equipment need to be cleared. Consequently, it is crucial for the workers to

Email addresses: aida.eduard@nu.edu.kz (Aida Eduard), dnislam.urazayev@nu.edu.kz (Dnislam Urazayev), aruzhan.sabyrbek@nu.edu.kz (Aruzhan Sabyrbek), daniil.orel@nu.edu.kz (Daniil Orel), dimitrios.zorbas@nu.edu.kz (Dimitrios Zorbas)

[1] https://ec.europa.eu/eurostat/statistics-explained/index.php?title=Railway_safety_statistics_in_the_EU

be notified before they can visually spot the approaching train. For instance, if the train is traveling at a speed of 60 km/h and the required evacuation time is 1 minute, the workers should be alerted when the train is at least one kilometer away from the repair site. This ensures that they have sufficient time to evacuate safely.

The proposed system comprises multiple components, each serving a specific function, as outlined in Section 3. To enable wireless communication, the system employs modern radio technologies, LoRa and ESP-NOW, using connection-less protocols. The former was chosen due to its advantageous characteristics, including long communication range, low cost, and high configurability. Additionally, ESP-NOW was selected for its compatibility with ESP32 devices, low cost, and ease of use.

To validate the effectiveness of the proposed mechanisms, the solutions were implemented and tested on real hardware. The designed prototype is a combination of commercially available equipment and custom-made devices. Importantly, the cost of implementing the proposed system amounts to less than $500, making it an affordable solution for any railway company in comparison with an ad-hoc TETRA radio system, which is currently used by many railway companies. For example, a set of two TETRA terminals is over $1500 without including other infrastructure costs.

The proposed system offers the following benefits: (a) cost-effective development and maintenance, (b) the ability for portable and ad-hoc deployment, (c) independence from pre-existing network infrastructure, (d) an individual worker notification mechanism through wearables, (e) notifications through both audio and vibration on wearables, (f) extended wearable battery life through a custom synchronized method, (g) support for up to four trains simultaneously, and (h) visualization of repair sites using Android and web applications.

In our previous work [3], we presented the general architecture of the system along with some early stage results. In this extended version, we briefly present the functionalities of the system with an emphasis on the communication aspects between the components as well as on their experimental evaluation. More specifically the following material is added:

1. A Machine Learning model is proposed and tested to estimate the distance between the station and a train when GPS data is not available.
2. The mathematical analysis for the case when the train coordinates are not available is extended taking into account the probability of receiving a beacon from the station device.
3. A multi-hop synchronized flooding mechanism based on ESP-NOW is extended to enhance the reliability and the battery lifetime of the wearable devices.
4. More detailed and accurate experimental results are presented for each of the components of the system.

The rest of this paper is organized as follows. Section 2 surveys and categorizes related railway notification systems suggested in the literature. Related work on distance estimation using LoRa is also presented in that section. Section 3 presents the proposed architecture and gives details about the functionalities of each component. Section 4 is devoted to the performance evaluation of the system using field experiments. Finally, Section 5 draws conclusions and ideas for future work.

2. Related Research

This section encompasses a discussion and a categorization of existing railway notification systems for different safety applications but it also surveys related works in the area of distance estimation using LoRa.

2.1. Related Railway Notification Systems

The proposed railway notification systems mainly involve stationary beacon devices sending data from sensors via cellular networks, or a set of AI-powered (Artificial Intelligence) visual and/or acoustic devices that detect objects, abnormalities, and moving vehicles. The following subsections provide a categorization of these approaches into visual, acoustic, and systems employing other types of sensors.

2.1.1. Visual systems

Visual surveillance with Computer Vision (CV) models is used for a vast range of applications, mainly involving human movement detection. The CV approach enhanced with AI for similar railway surveillance applications has also been investigated by researchers in the literature.

Leone et al. [4] proposed the use of an ARM-powered board equipped with a night vision camera to detect instabilities leading to landslides and other obstructions – for example people – on the rail tracks. The results showed that rocks falling on the railroad and passing people are clearly seen on the tracking area of the device.

Systems that detect objects in railways have been proposed for more than a decade. Oh et al. [5] presented a system consisting of an array of cameras each of them covering up to 20 meters of tracks. Upon successful detection of a person on tracks, the control room of the station is alarmed. The authors do not specify what kind of communication system is used for alarming the station staff. In a similar context, Alessandretti et al. [6] developed a system used for vehicle detection on the road, a system that could be used for train detection as well. On the contrary, instead of detecting specific objects on the railroad, the system proposed by Mukojima et al. [7] detects only foreign objects on the railway, using a subtraction techniques by comparing actual frames to reference frames, eliminating the need to train a ML model on vast types of objects.

A CV approach was also followed by Sikora et al. [8]. In their system, called AISS4RCT, a set of powerful boards send detection data with an achieved average recall of 89% by using the YOLO model. Their system is able to evaluate the state of the traffic lights on the crossroads and the position of a railroad barrier. These abilities enable AISS4RCT to be used for regular traffic, as well as for the railroad junctions, and thus, enhance safety.

The latest monitoring techniques often utilize Light Detection and Ranging (LiDAR) technology, which offers a significant advantage due to its long detection range. However, drawbacks of LiDAR equipment are its large size, high cost, as well as being vulnerable to extreme temperatures and precipitation such as ice and snow. Some additional drawbacks include the need for a permanent network infrastructure, the potential for negligence in supervision as discussed by Baysari et al. [9].

Taking this a step further from typical camera-only and LiDAR-only approaches, Zhangyu et al. [10] proposed a system that combines both a camera and a LiDAR component. LiDAR is complimenting the depth-perception capabilities of the system, which regular camera modules cannot provide. This fusion method allowed their model to outperform other systems that use only one of the two components.

Even though many of the aforementioned systems seem to work well in real conditions, they either require permanent infrastructure near the rails or their cost is high. Moreover, some of them require seamless power supply and the presence of specialized personnel to operate the system.

2.1.2. Acoustic systems

Apart from the visual inspection, acoustic data can also be used to enhance railroad safety. Acoustic methods for accident prevention employ sensors that detect noise and vibrations either on the rails or in close proximity to them, as explained in the study of Sato et al. [11]. Microphones in the system are fixed near the railroad and are used to collect ambient noise coming from approaching trains. Using logistic regression they achieved a high accuracy with an F-score of 0.987.

Acoustic analysis of the ambient noise is an approachable nearly non-intrusive way to detect passing vehicles, as it was also shown by Ishida et al. [12]. The authors exploited the difference in noise profiles of two identical microphones placed at different distances to define the direction and the speed of the vehicle using the sound mapping technology, achieving an F-score of 0.92. Considering the simplicity and structure of the system, it is highly mobile, and can be relocated if required.

A more recent development in this field is the Distributed Acoustic Sensor (DAC) technology, which makes use of fiber optics as exceptionally receptive sensors for detecting vibrations and acoustic signals associated with train arrivals. Dumont et al. [13] describe a system that uses fiber optics and the principle of Rayleigh Backscatter, by sending laser light pulses and capturing the refracted light. To analyze data obtained from DACs, the authors came up with a deep learning approach, whereas Milne et al. [14] decided to follow a mathematical approach to use the phase-sensitive Optical Time-Domain Reflectometry (φ-OTDR) interrogation. Overall DAC solutions are very effective in terms of latency, data-rate and accuracy, however, they require a fiber cable to be deployed across the entire railway which is cost- and labor-inefficient.

2.1.3. Other systems

Since AI-based approaches often involve powerful machines and expensive equipment, there are attempts to build systems without using AI. Accelerometers were used in systems proposed by Khivsara et al. [15] and Shrestra et al. [16], although these sensors were used differently in these papers. In the case of the train-mounted system designed by Khivsara et al., the accelerometer is used to determine the state of the train, whether it is stationary or moving. In the case of an abrupt deceleration or long inactivity, a message is sent using a GSM radio technology (Global System for Mobile Communications). On the contrary, Shrestra et al. [16], connected an accelerometer to the rail track, reading the vibration data caused by approaching trains. Additional sensors are installed on tripods several meters away from the track, to gather speed, direction, length, and other information about the train approach characteristics. This information is sent to the maintenance teams located not very far from the sensor system. The solution does not require a permanent infrastructure, but the deployment of the system is not instantaneous. Preliminary installation of the sensors near the railway is required.

In [17], Jain et al. express their concern of accidents happening due to human error, and by eliminating it, they proposed a system aimed to increase safety of railway crossings. The authors proposed an infrared (IR) sensor system to determine if a train passed a certain stationary point. This information is then forwarded to an actuator to open or close barriers.

Fu developed their train approach notification system using GSM-R (Global System for Mobile Communications – Railway) and GPS functionality [18]. Since the data is sent to a remote server, it can be accessed anywhere. It is highly efficient, but requires an existent GSM-R network, which may not be always available.

In general, the above-mentioned systems are mostly stationary and rely on the use of existing infrastructures, such as a cellular network (e.g., GSM). Our proposed system is designed to be ad-hoc without requiring permanent access to external networks.

2.2. LoRa-Based Distance Estimation Models

LoRa-based distance estimation models leverage the Long Range (LoRa) communication technology to accu-

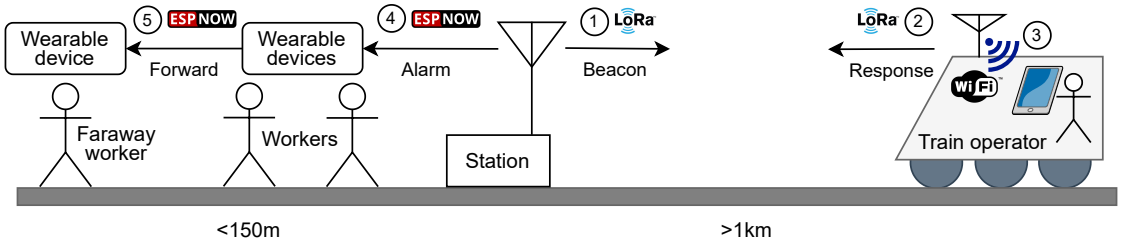


Figure 2: Overview of the system components and basic communication steps.

rately measure distances between devices in a wireless network. These models utilize characteristics of LoRa signals, such as received signal strength (RSS) and airtime, to estimate the spatial separation between nodes. By harnessing the low-power, long-range capabilities of LoRa, these distance estimation models find applications in various fields, including smart agriculture, asset tracking, and smart city environments. However, as it can be observed by the following paragraphs the conclusions are mixed, even though more studies favor the use of ML models compared to path-loss ones.

The first study by Dieng et al. [19] introduces a dynamic RSS-distance mapping for animal herd localization using collar devices. It employs LoRa beacons and GPS-equipped nodes, utilizing a log-distance path-loss formula. The Root Means Squared Error is reported to be 0.77 meters.

Rademacher et al. [20] explore path-loss models in urban conditions, favoring the log-distance model over terrain models. DeepLoRa, proposed by Liu et al. [21], utilizes deep learning and satellite images for long-distance path-loss prediction. However, its reliance on expensive multi-spectral images poses challenges.

Radeta et al. [22] assess LoRa network performance in marine environments, employing gateways and end-nodes on sea vessels. Demetri et al. [23] combine SVM and path-loss models with multispectral images for link quality estimation, using the Okumura-Hata model.

Lam et al. [24] propose two LoRa localization algorithms using the log-distance path-loss model in urban settings. Anjum et al. [25] compare positioning models, with the smoothing spline model showing the highest accuracy outdoors. Islam et al. [26] propose ML distance-mapping using RSSI, SNR, and SF parameters, achieving the best accuracy with the random forest model.

Finally, Carrion et al. [27] compare ML models (RFR, NN, LSTM) using simulated RSS-GPS coordinates, acknowledging differences from real data. These studies collectively explore diverse approaches to enhance localization and path-loss modeling in various environments, each with its unique challenges and innovations.

3. System Architecture & Functionalities

In this section, we provide an overview of the proposed system architecture and its components, with a main focus on the communication aspects between these components.

3.1. System Overview

A simplified representation of the system is depicted in Figure 2. The system consists of three components: the ad-hoc station, the train, and the worker components. Additionally, an optional back-end system is included to facilitate the interconnection between the components and provide connectivity to an external network. The back-end system's external network is employed only when an Internet connection is accessible, while it is not necessary for the workers' notifications. More details about the back-end system are provided in [3].

A general description of how the system works is given below, while each of the system's components is described in more detail in the following subsections. It is important to note that, for simplicity, the discussion assumes the presence of a single station device, although multiple stations may exist at the same time.

The station initiates periodic transmissions of beacons to broadcast its location and notify incoming trains about its presence. Upon receiving these packets, trains respond by providing their own coordinates, their speed, and their direction. The station receives these individual responses and calculates the estimated arrival time of each train. This information is then forwarded to the workers' wearables, allowing them to be notified about approaching trains. As it is depicted in Figure 1, the devices on trains can also provide a real-time visual notification to the train operators by forwarding received station location to onboard equipment (tablets).

3.2. The Portable Station component

In order to gain a deeper comprehension of the train and workers components, we will commence by elucidating the portable station component which functions as an intermediary linking these two elements.

The station component consists of two Micro-Controller Units (MCUs) and a positioning system module such as GPS. The first MCU is responsible for the communication with the train component over the LoRa technology, while the other MCU handles the communication with the workers component, employing ESP-NOW (or WiFi). They are connected with each other via an Inter-Integrated Circuit

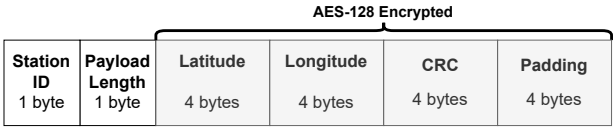


Figure 3: The station beacon packet format.

(I2C) link. The presence of two MCUs is practical and allows for flexibility and seamless connectivity between multiple components of the system. The station device can be powered by batteries or a support vehicle. It exhibits a low power consumption which allows seamless operation of several days with a typical 10Ah powerbank. Throughout this paper, it is assumed that the station is not an energy constrained device.

The train and station components exchange GPS coordinates to estimate the arrival time and notify the workers. The first MCU of the station periodically transmits beacons to broadcast its location. The format of the beacon, as shown in Figure 3, includes an encrypted payload using the keys that were shared preliminary. A train receives the beacon from the station upon entering its working range, and then responds with its own packet of current GPS coordinates along with the speed and the direction. The great-circle distance is then calculated based on the GPS locations of both devices using the Haversine formula. Each station has the capability to support up to four trains simultaneously approaching from different directions. To prevent collisions in scenarios where multiple trains exist in a station's neighborhood at the same time, one out of four timeslots is chosen after receiving a beacon. The selection of a timeslot is described in the next subsection.

Upon receiving the data from the train, the station's first MCU uses the train's response data to estimate the train's arrival time. If no positioning data is available, the station device estimates the position using the response RSS value. By employing one of these methods, the station component can provide workers with timely information about the approaching train, enabling them to take appropriate safety measures.

3.2.1. Machine Learning distance estimation model

As it was mentioned in the previous paragraph, the RSS value of the response packet is employed in the case where the train packet does not contain positioning data or if the data is corrupted. The estimation is not of critical importance because the workers will evacuate the site anyway once they receive a message from the station device. However, it can be used as a metric to identify the speed of the train – and thus its arrival time – when positioning data is not available.

In this scenario, we leverage ML to automatically map the RSS to an estimated distance. The reason of using ML is to better adapt to the deployment conditions in terms of terrain type compared to the conventional method of approximating the distance using mathematical equations

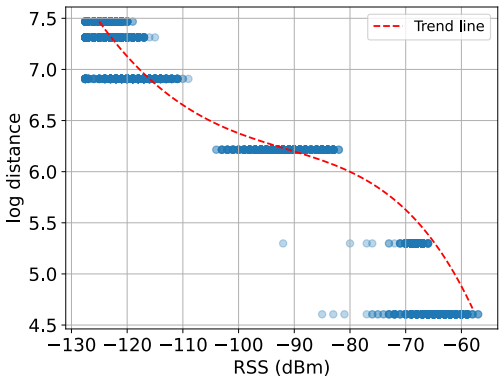


Figure 4: Dependency between natural logarithm of distance and Received Signal Strength.

that represent the actual path-loss [28, 3]. In such a model, the distance is calculated by a mathematical formula where it is necessary to know the path-loss exponent, the reference distance, and the path-loss at that distance. These three parameters can be learned experimentally but the model would work only for the specific conditions under which the experimental data was collected [29].

In our approach, using experimental data from different conditions, we can select ML coefficients that can map the RSS value and the information on the terrain type to distance, minimizing the error of predictions. Similar approaches have been followed in the literature [21, 30]. The system can autonomously identify the surrounded terrain type and adapt the model accordingly by loading the corresponding coefficients that are closer to the identified terrain type. The identification of the terrain is done using pre-loaded maps on the station device and the "Corine Land Cover" functionality of a Geographical Information System. How this system works is out of the scope of this paper. Currently, 3 types of terrains are supported; 0 for Line-of-Sight (LoS) and no obstacles, 1 for LoS and few obstacles (e.g., people and short trees), and 2 for non-LoS and severe obstacles (e.g., hills, buildings, high trees, forests).

To decide about the ML coefficients, the data is split in 80-20 proportion on training and testing sets. Then, the parameters of the Ordinary Least Squares Linear Regression model are calculated on the training set. To evaluate how well the model performs on the training data, we obtained its prediction on the test set and calculated the Mean Absolute Percentage Error (MAPE) [31].

To tackle the skewness of data Quantile Regression is employed, which goes beyond predictions of the mean values and allows targets and the features to vary across the conditional quantiles of the data distribution [32]. This is needed in order to better fit the median of our data. As a result, the trend among RSS-distance points can be modeled using a third degree polynomial function, as it can be seen in Figure 4. Using Quantile regression and polynomial features, MAPE is reduced to 17.25%. Applying this model to the training data for terrain type 1, we

Figure 5: A TETRA terminal (left) equipped with Android software to communicate with a station device (center), and size comparison with a 2€ coin (right).

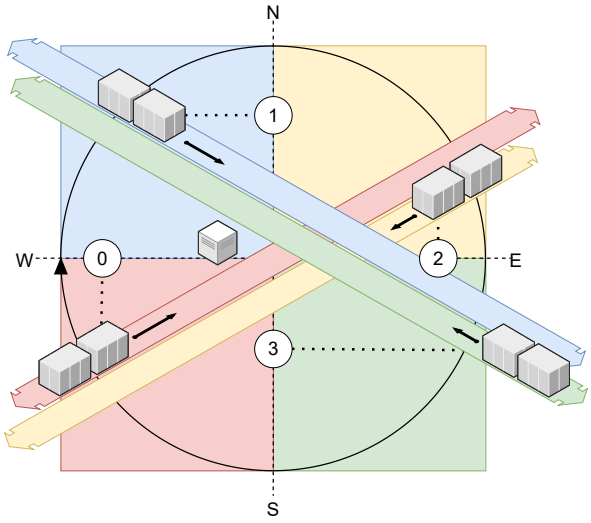


Figure 6: The train selects the timeslot based on its direction relative to the station. An illustrative example of all slots being occupied.

get an average error of 120 meters. The model's efficiency is further assessed in Section 4, where we do experiments using different distances between the station and the train devices that were not used for training the model.

3.2.2. Application layer

The second MCU of the station can alternatively be used as a WiFi access point to connect Tetra network terminals (see Figure 5). Tetra networks are usually used by railway companies to provide intercommunication between trains and stations but they require a permanent infrastructure. In cases where a Tetra terminal is available in the workers team, the administrator can set the station device to send the alerts to a Tetra device instead of the workers' wearables. An Android app running on those terminals has been developed for this purpose. The same app can also be used to visualize the train positions on the TETRA terminal.

3.3. The Train component

The train component is a sub-system that responds to the beacons transmitted by the portable station but it can also communicate with train operators using another Android application which can alert train operators about the approach to nearby repair sites [3]. The communication functionalities of the train component play a crucial role in the overall system, as explained in the following paragraphs.

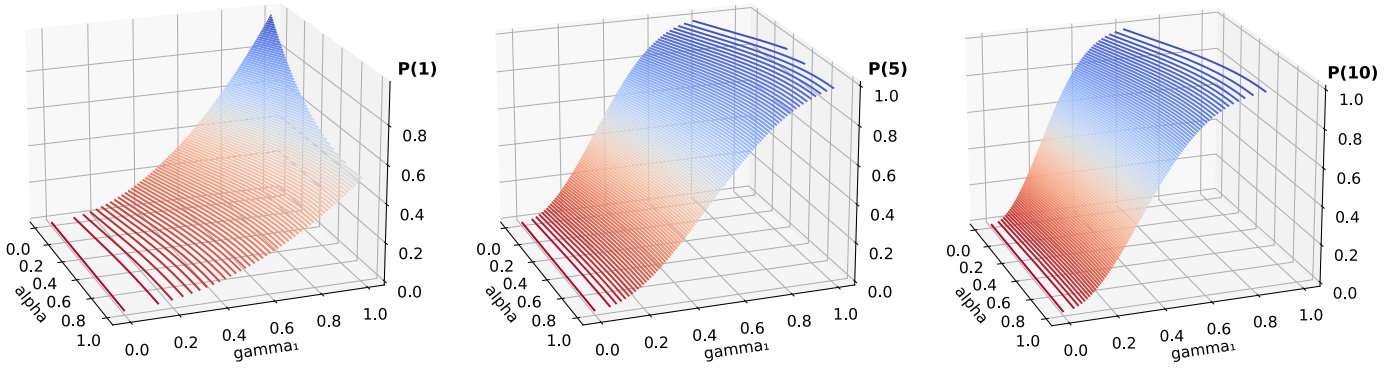
The train's response packet follows a similar format with the station beacon. The only difference is that the train speed and the direction are also included in the response packet along with the GPS coordinates. This information is encrypted using pre-shared keys and it is put together with a unique train identification number and the payload length, as depicted in Figure 9. Before responding to the station, every train must determine which timeslot should be selected to transmit the packet. The selection process relies on the direction of the train's movement, in a clockwise order as shown in Figure 6. For instance, a train moving from southwest would choose slot 0 and a train coming from northwest would choose slot 1. The system is designed for rural areas far from stations where up to 4 trains can arrive at the same time, even though 2 trains would be enough for most of the countries[2]. Thus, we consider a worst case scenario as with four tracks of trains coming from different directions. Because trains are coming from four directions, it is safe to assume that 4 slots based on the cardinal directions (N-E-S-W) will be sufficient.

The length of the window slot is determined by the LoRa settings and the number of transmitted bytes. For example, with 18 bytes of payload and SF10 taking into account some clock drift and processing time, a timeslot is approximately 350 milliseconds long. Therefore, from the moment the train receives a beacon, the communication delay cannot exceed 1.75 seconds.

3.3.1. Multi-slot probabilities

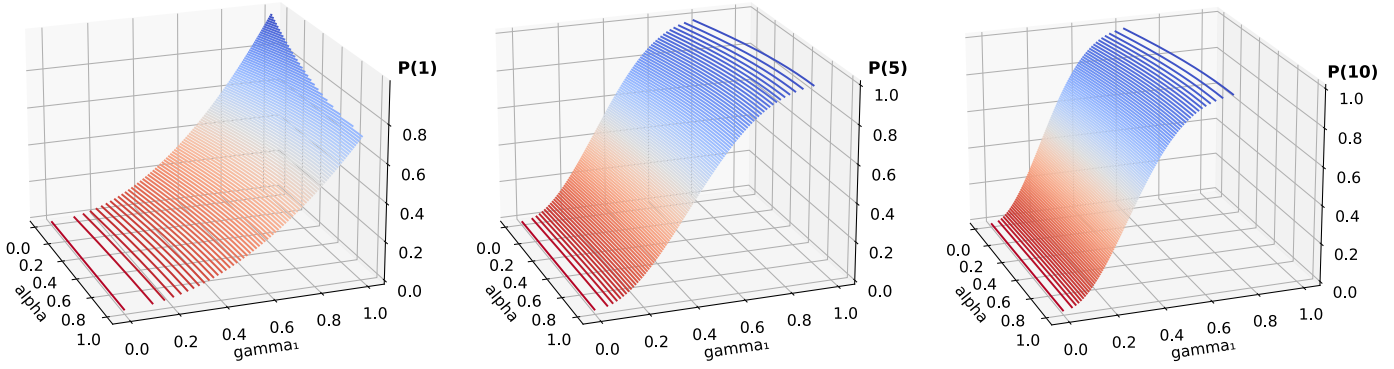
In situations where the train's coordinates cannot be obtained, such as due to malfunctions or adverse weather conditions, the response packet will contain null coordinates and will randomly select one timeslot at random from a set of N slots, where $N > 4$. The additional slots are used to reduce the probability of collision when two trains choose the same slot. As shown in the analysis below, if additional slots are used, this probability reduces

[2]It is worth noting that the proposed system does not account for acute crossings, as these are uncommon in the rural areas where the system is intended to be deployed. These scenarios are primarily located close to train stations that are already covered by regular networks, making the proposed system unnecessary in those cases. However, even if this happens, the probability of packet collision is extremely low because one of the two train responses will still be received by the station and the railway will be evacuated anyway.



(a) t = 1 (b) t = 5 (c) t = 10

Figure 7: Probability of "receiving a station beacon, any single train does not select the same slot, and the response is received at the station" for different α values (probability the GPS is unavailable), γ_1 values (probability a station beacon is received, number of received beacons ($t \in [1, 5, 10]$), and $N = 5$ (timeslots).



(a) t = 1 (b) t = 5 (c) t = 10

Figure 8: Probability of "receiving a station beacon, any single train does not select the same slot, and the response is received at the station" for different α values (probability the GPS is unavailable), γ_1 values (probability a station beacon is received, number of received beacons ($t \in [1, 5, 10]$), and $N = 10$ (timeslots).

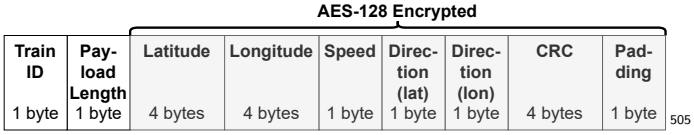


Figure 9: The train response packet format.

substantially. The number of additional slots that could be reserved depends on the speed of the train and how often the beacons are transmitted. However, it is always important to use the minimum possible slots to reduce the maximum response time in the system.

Using the analysis shown in [3] and assuming a constant train speed, the probability of a response packet not colliding with other packets after t beacons ($t \geq 1$) is:

$$\mathcal{P}(t) = 1 - \big(1 - \mathcal{P}(1)\big)^t. \tag{1}$$

This is because every transmission period represents a unique occurrence, implying that the likelihood of a non-collision event in two separate periods is unrelated. Consequently, the probability of encountering at least one collision in t transmission periods can be calculated as t multiplied by the complement of the probability of no collision, which is $(1 - \mathcal{P}(1))$.

Note that the probability of selecting a single timeslot without collisions (i.e., $\mathcal{P}(1)$) is linearly dependent to the average probability of receiving a beacon by a train within a fixed time period (denoted with γ_1), the probability of selecting a slot that does not collide with other packets (denoted with $\mathcal{A}$), and the average probability of successfully receiving the response at the station within a fixed time period (denoted with γ_2). As the train is moving towards the station, $\gamma_2 \geq \gamma_1$. Thus, $\mathcal{P}(1)$ is computed as follows:

$$\mathcal{P}(1) = \gamma_1 \cdot \mathcal{A} \cdot \gamma_2 \geq \gamma_1^2 \cdot \mathcal{A}, \tag{2}$$

where $\mathcal{A}$ is given by the following formula for N available

slots:

$$\mathcal{A} = \frac{3\alpha^2 - 6\alpha}{N} + \frac{9\alpha^2 - 6\alpha^3}{N^2} + \frac{3\alpha^4 - 4\alpha^3}{N^3} + 1, \quad (3)$$

where α is the probability that the coordinates are not available.

Figures 7 and 8 illustrate $\mathcal{P}(t)$ for various combinations of received beacons, α, and γ_1 values when N=5 and N=10, respectively. We can observe the increase of success probability as the number of slots increases. Moreover, it is noteworthy that the probability with just one received beacon (i.e., $\mathcal{P}(1)$) is relatively low. However, as the number of received beacons increases, this probability approaches 1.

To illustrate that, let us consider an example scenario where (i) trains can receive beacons up to a maximum distance of 2 kilometers from the station, (ii) time constraints allow the last beacon to be received within 1 kilometer from the station, and (iii) the beacon rate is set at 1 beacon every 5 seconds. In this scenario, a train can receive a maximum of 12 beacons within the specified time period (i.e., 60 seconds assuming a train speed of 60km/h). Assuming that the average packet reception rate in that time period is 83.33% and α is 1 for all trains, the probability tends to 1. Apparently, the example can be adapted to other scenarios given a different speed of trains and beacon rates.

3.4. The Workers component

The system assumes that the wearable devices are securely fastened to the wrists of all or some of the workers at the repair site. The wearable device comprises an MCU with an IEEE802.11 transceiver, an OLED display, a beeper, a vibration motor, and a power unit. Technical design details of the wearable devices are shown in [33]. The wearable devices receive notifications about approaching trains from the station device and notify the worker through haptic mechanisms. A beep speaker generates short repeating beeps to indicate the proximity of the train, while the vibration motor provides notifications through vibration. The device display shows the direction and the arrival time of a train. The focus of this paper is on the communication part between the station and the wearables.

3.4.1. The Always-on approach

The train arrival data is promptly transmitted to the workers' wearables using an ESP-NOW-based protocol. The primary goal of this protocol is to ensure a high level of reliability (>99% delivery ratio) for all devices. The objectives of the protocol are (a) to minimize the response time once a train is detected, (b) to provide high reliability, and (c) to exhibit a battery life of a full working shift (8-9 hours). The first objective is achieved by allowing the wearable devices to continuously be in listening mode, enabling them to receive incoming packets without delay. This ensures that workers are promptly notified as soon as a train's response is received, facilitating timely and efficient communication within the system. The second objective is achieved using successive retransmissions and a multi-hop mesh network. This functionality is described in the next paragraphs. The third goal is demonstrated in Section 4, where we show that the wearable can achieve a lifetime of approximately 10 hours in the Always-on mode.

As depicted in Figure 10A, we employ a mesh architecture together with multiple successive transmissions and multi-hop flooding. More specifically, each packet initiated by the station is sent k consecutive times. Once a device receives a packet from the station, it switches to transmit mode, and it forwards it to other devices within its range (also k times). As it is shown in Section 4, this action does not have any effect on the energy consumption as only 1ms of time is required for a couple of additional transmissions while the additional communication cost is not high either. By employing this multi-hop forwarding mechanism, the protocol enhances the overall reliability and coverage of the communication system allowing even distant devices to reliably receive beacons. Moreover, it provides an additional opportunity for devices that may miss the station's packet to receive it from other nearby devices. The number of maximum hops n is decided based on the application requirements in terms of coverage and reliability.

3.4.2. The Synchronized ESP-NOW approach

Sync-ESP-NOW [34] is proposed as a synchronized 2-hop mechanism to enhance energy efficiency by allowing the end-devices to stay in deep sleep mode for some period of time. The approach works over the ESP-NOW link layer developed by Espressif for IEEE802.11 transceivers and it is quite popular for connection-less peer-to-peer communications in industrial and home automation applications [35, 36]. The purpose of using Sync-ESP-NOW in this project is to provide an alternative solution for scenarios that may require much higher battery lifetime than the Always-on approach. In Sync-ESP-NOW, the station broadcasts packets in a synchronous way and in short periods, allowing the wearable devices to spend some time in deep sleep mode and wake up in predefined time intervals to receive data.

In this paper, Sync-ESP-NOW is extended to allow devices to send up to k successive packets over n hops. The frame format is changed radically compared to the previous version in order to accommodate these two functionalities and it is presented in Figure 10B. The station transmits periodic beacons containing train information as depicted in Figure 10C. The packet contains several fields, including the sequence number, which serves as a unique identifier for the transmitted packet. The most recent sequence number is stored by each end-device to avoid processing duplicate downlinks. The maximum number of packet forwardings by the wearable devices is represented by the Time-To-Live (TTL) value. Initially, it is set to n at the station, and upon reception, each receiver decre-

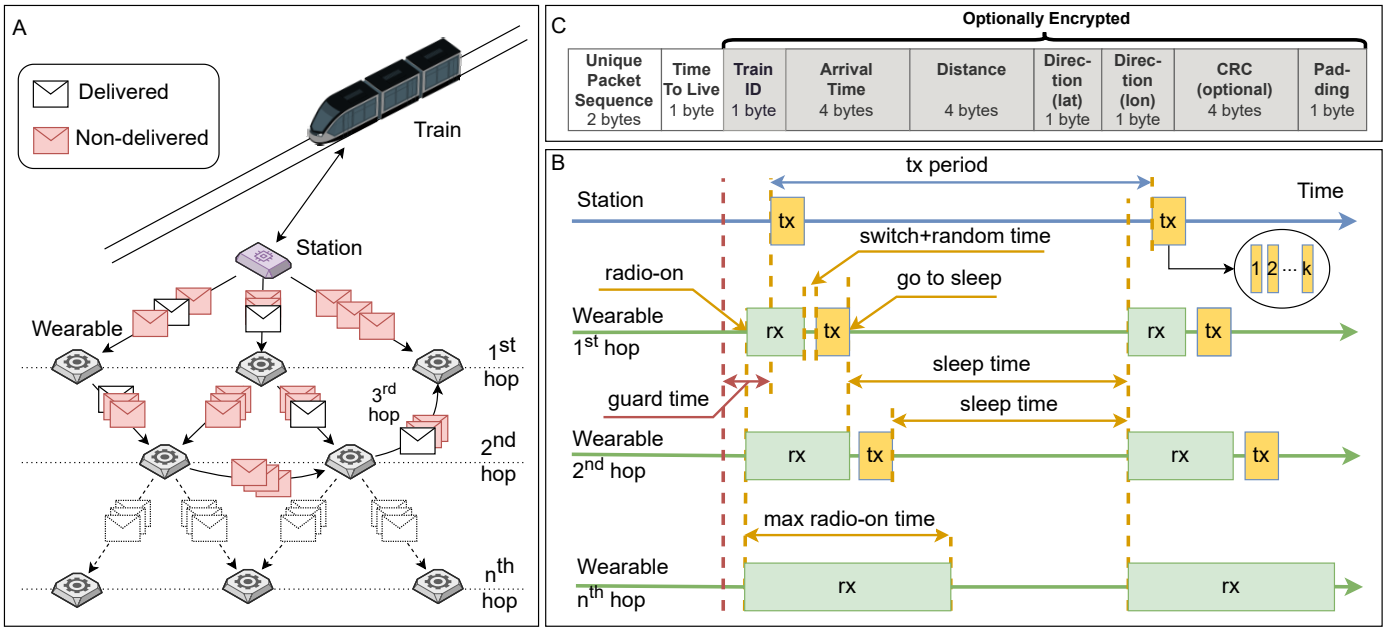


Figure 10: (A) The multi-hop functionality of the approaches. Each unique packet is transmitted k consecutive times and it is forwarded to all neighbors up to n times. (B) Timeframe of Sync-ESP-NOW at different hops; (C) Packet format of the approaches.

ments it by 1. If the TTL of the packet is greater than 0, it is then forwarded to other devices within the range. Additionally, there is an option to encrypt the packet payload with pre-shared keys, providing an additional layer of security for the transmitted data.

The wearables use these beacons for clock synchronization purposes as well. Hence, the station transmits such a packet regardless the presence of a train. If no train is approaching, a train id equal to 0 is used, followed by null data for the arrival time, the distance, and the directions. Synchronization is achieved by allowing the devices to wake-up some milliseconds earlier than the expected transmission time (guard time). The guard time is used to compensate for any clock drift. The amount of time the radio stayed on compared to the arrival of the beacon, determines the clock correction that is applied. Each wearable stays in receive mode (rx) for a max predefined time (max radio-on time in Figure 10B). Once it receives a beacon it checks the TTL value and it switches to transmit mode (tx) to forward the beacon to neighboring devices. The max radio-on time is a function of k, the guard time, the radio mode switch time, and a random time used to mitigate overlaps of neighboring transmissions. Assuming that each beacon is about 20bytes long, $k = 3$ (tx time = 4ms), a guard time of 40ms, a switch time of 5ms, processing time of 10-20ms, a max random time of 1ms, and n=3 hops, the longest time a device can spend in receive mode is 180ms (40+(3-1)(40+20+4+5+1)). If the device does not receive a beacon within this time limit, it adjusts its sleep time accordingly to receive a beacon in the next round. Two successive synchronization fails cause the device to switch to the 'Always-on' mode until it receives one of the next beacons.

Table 1: Experiment Parameters & Radio Settings

Parameter	Value
Train/Station/Wearables devices	2/1/8
LoRa Spreading Factor	10
Radio channel (duty cycle)	869.525 MHz (10%)
Channel bandwidth	125 kHz
Code rate	4/5
Preamble symbols	10
Transmission power	14 dBm
Station packet length	18 Bytes
LoRa Explicit header / CRC	No / No
ESP-NOW Mode	LR (250 Kbps)
Wearable packet length	17 Bytes
Wearable transmissions (k)	1 and 3
Time-To-Live (N)	1 and 2
Guard time	35ms
Encryption	AES-128
Sync-ESP-NOW delay	5 sec

Sync-ESP-NOW achieves up to 96% lower energy consumption compared to the always-on approach but it adds some delay to the notification of the workers. The delay depends on how often the wearables wake up to receive data from the station. The longer the sleep time, the higher the energy savings but the longer the delay. However, as it was shown in [34], even a second of sleep time can extend battery lifetime by more than 300%.

4. Evaluation & Discussion of the results

The proposed system was implemented on open hardware using ESP32 devices and detailed schematics are presented in [33]. The system underwent a comprehensive evaluation, both as a unified system and as individual components. In order to present the results effectively,

Table 2: Packet Delivery Ratio of the Station-Train communication at different distances.

Distance (m)	Avg. RSS (dBm)	PDR (%)	Repeated Losses
500	-102.74	98.8	1
1000	-119.93	96.4	2
1750	-121.54	81.4	3

the evaluation outcomes are provided separately for each component. The parameters used in the experiments are summarized in Table 1, outlining the key variables and settings employed during the evaluation process.

4.1. Station-Train Link

The evaluation of the train-station communication included multiple field testings. They were conducted using one station and two train devices across an open space spanning a few square kilometers. The geographical features consisted mainly of optical contact between devices but with the presence of light obstacles such as tree leaves and people. The experiment's primary objective was to determine how different environmental conditions affect the path loss.

The results indicated that the maximum effective range[3] that was achieved was 5.7 kilometers with LoS. In scenarios where LoS was obstructed, such as in areas shadowed by buildings or within forests, the range was limited to no more than 500 meters. In those situations, an intermediate device as the one presented in [16] must be placed not far from the station to act as the repeater or the station must be elevated with the help of wired drones [37] or balloons [38]. These solutions are being investigated but are currently out of the scope of this paper.

Table 2 displays selected results obtained under LoS conditions with the presence of light obstacles. The findings reveal that even at a distance of 1.75 kilometers, the Packet Delivery Rate (PDR) remains acceptable, with only 3 instances of successive packet losses at worst case. For example, given a beacon interval of 5 seconds, a train speed of 60km/h, and one minute evacuation time, a train can receive more than 10 beacons per km. Thus, even with 3 successive losses, the station will finally receive a response from the train before the hard constraint of 1 minute (or 1 km). These results also demonstrated the system's ability to maintain a reliable communication even when faced with moderate obstacles in the transmission path and over significant distances.

The system was also tested for its response time which corresponds to the round-trip time between the station and the train plus extra processing time at the station before forwarding the packet to the workers. The results are depicted in Table 3 and reveal – as it was expected – a linear increase of the response time. The worst response time was measured at approximately 2.3 seconds.

[3]We consider a range as effective if it exhibits a packet success rate of at least 66.7%.

Table 3: Response time results when the number of timeslots is 5 (N=5).

Slot number	Response time (sec)	
	Average	σ
0	0.904	0.009
1	1.211	0.011
2	1.560	0.007
3	1.911	0.008
4	2.260	0.007

Table 4: Machine Learning model distance estimation results.

Distance (m)	Path loss (dB)	σ RSS (dBm)	Model (m)	δ to real distance (m)	MAPE (%)
100	76.45	3.77	148.65	+48.65	48.65
500	103.657	4.36	506.97	+6.97	1.39
1000	131.186	3.64	1239.87	+239.87	23.99
1200	131.08	2.12	1197.33	-2.67	0.22
1400	129.95	2.51	1131.55	-268.45	19.18
1500	134.71	2.51	1521.44	+21.44	1.43
1600	136.542	2.29	1712.99	+112.99	7.06
1750	135.155	1.65	1547.64	-202.36	11.56

4.2. Machine Learning RSS model performance

To ensure that workers are notified for incoming trains, regardless of any potential GPS malfunctions or packet corruption, an RSS model was developed using Machine Learning methods discussed in Section 3.2. A number of experiments were conducted to collect data and train the model as it was described in Section 3.2. In every setup, 500 packets, each consisting of 20 bytes, were transmitted from the station device to the train device and the RSS values were recorded. All experiments were conducted on a flat terrain in Kazakh steppes with a few obstacles present between the receiver and the transmitter (terrain type 1). The distance computed by the model was compared to the real distance. Mean Absolute Percentage Error (MAPE) serves as a metric for assessing the model's performance. This error quantifies the percentage by which predictions diverge from the actual values.

The results of these experiments are summarized in Table 4. They show that the model predicts the distance with a maximum δ of approximately 270 meters , equating to a deviation of 19.18% from the actual distance. However, the highest MAPE, nearing 50%, was observed for a distance of 100 meters – the point at which the train should be distinctly visible. Assuming our use-case scenario with a train speed of 60km/h, the highest absolute deviation is translated to an error of maximum 17 seconds. We can also observe that for distances above 1200 m the model gives more accurate predictions than for ones below this value. Also, since the model tends to underestimate distances, the deviation in terms of predicted and actual time is not that huge.

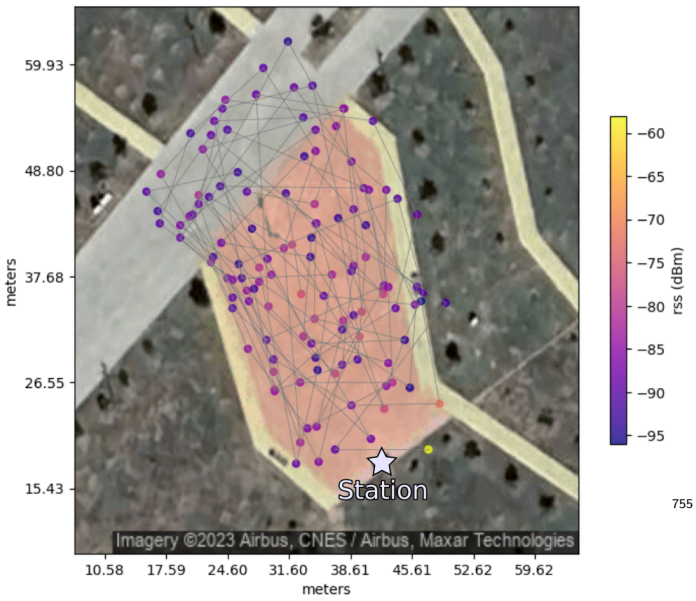


Figure 11: Wearable experiments site: dimensions, mobile positions, and their corresponding RSS values.

Table 5: Packet reception ratio (PRR) and received signal strength (RSS) for different positions and TTL values (1 transmission).

TTL	Experiment	RSS (dBm)		PRR (%)		
		Avg	σ	Total	1st hop	2nd hop
2	**Always-on**	-80.65	8.06	99.95	88.0	12.0
	Sync-ESP-NOW	-80.81	1.23	99.46	87.2	12.8
1	**Always-on**	-78.84	8.53	90.75	100	–
	Sync-ESP-NOW	-79.84	3.73	90.42	100	–

4.3. Station-Wearable Link

To test the station-wearable communication, experiments were conducted in an outdoor environment of 1500 square meters size, characterized by numerous light obstacles including trees and people as it is depicted in Figure 11. The experiments were designed to assess and compare the performance of the regular ESP-NOW approach to the synchronized method, with the multi-hop forwarding mechanism disabled or enabled. Moreover, in order to test the effect of multiple successive transmissions, the experiments were repeated with k=1 and k=3. Eight devices – most of them mobile – were employed for these experiments. Traces of one of those devices along with the corresponding RSS values are depicted in Figure 11.

Table 6: Packet reception ratio (PRR) and received signal strength (RSS) for different positions and TTL values (3 transmissions).

TTL	Experiment	RSS (dBm)		PRR (%)		
		Avg	σ	Total	1st hop	2nd hop
2	**Always-on**	-79.21	7.78	99.98	88.86	11.14
	Sync-ESP-NOW	-77.37	4.10	99.99	93.35	6.65
1	**Always-on**	-82.05	5.87	98.18	100	–
	Sync-ESP-NOW	-80.53	5.73	95.73	100	–

Table 7: Average power consumption per round for different TTL values (n) and successive transmissions (k).

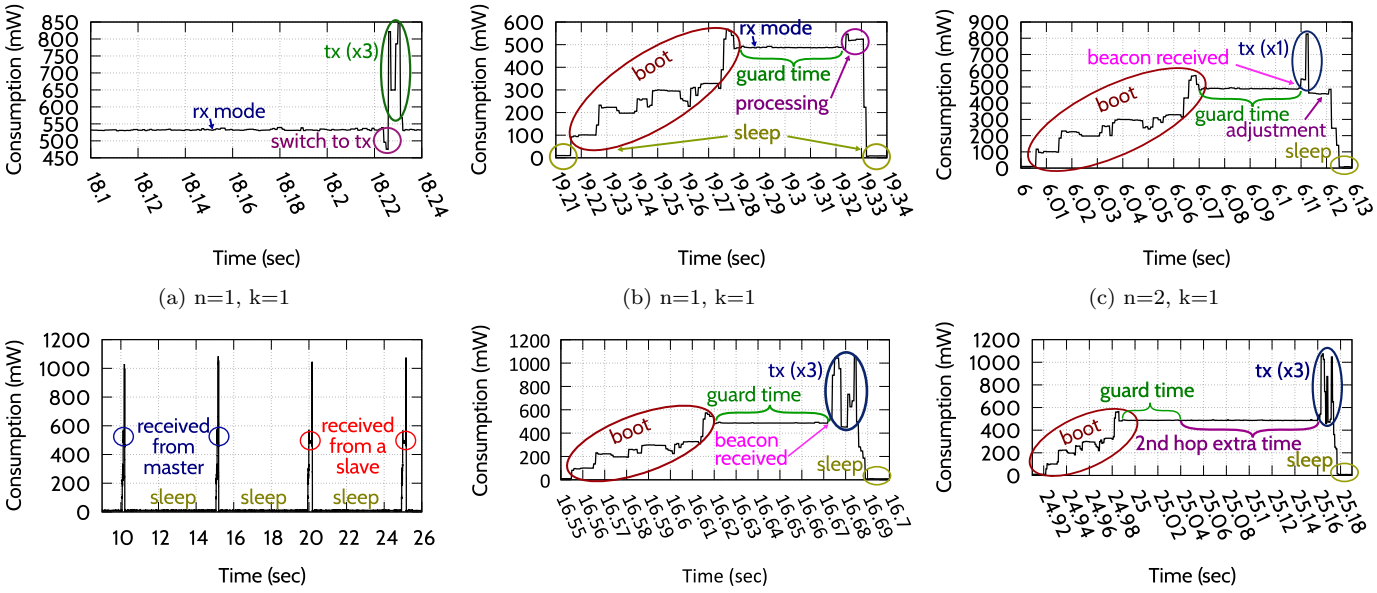
Approach	n	k	Average power consumption (mW)	Received from hop
Sync-ESP-NOW	1	1	18.0	1st
	1	3	18.1	1st
	2	1	20	1st
			29.1	2nd
	2	3	20.2	1st
			31.8	2nd
Always-on	1, 2	1–3	533	1st, 2nd

The results can be found in Table 5 for $k = 1$ and in Table 6 for $k = 3$. First of all, comparing the results of the two tables, we can witness the effect of multiple successive transmissions on PRR for the single-hop scenario. PRR increases from approximately 88% to more than 95%. Regarding the synchronized approach, the number of desynchronizations was only 1. Moreover, looking at the two tables individually, we can observe that even a second hop is enough to provide an over 99.4% PRR. The last two columns reveal the high contribution of the second hop. The multi-hop mechanism combined with multiple successive transmissions gives an almost 100% PRR for this series of experiments.

4.4. Power Characterization

A series of experiments were also conducted to assess the amount of power required for the operation of a wearable device but also to compare different modes of operation. The power consumption was measured using a power analyzer and the results are depicted in Table 7. The results show an average power consumption of 533mW per round in the Always-on mode. A 1500mAh battery can provide approximately 19000 Joules of energy. This is translated to almost 10h of lifetime (i.e., (19000/0.533)/3600). This number was later confirmed by an experiment, where a wearable device was periodically receiving alerts every 30 minutes with a 2-min period of continuous haptic notifications (visual, sound, and vibrations) until it runs out of battery. The measured battery life of the prototype exceeds the application requirements, as a typical work shift does not exceed 8-9 hours. By using Sync-ESP-NOW and a wake-up time of 5 seconds, it is possible to decrease the average consumption to 18-32mW (94% decrease), and thus, increase the battery lifetime to 4.5-5 days. However, as it was mentioned earlier in the text, this comes with an additional response delay of 5 seconds. Nevertheless, even an 1 second of sleep time can significantly increase the lifetime of the wearable. The sleep time can be adjusted according to the specific application requirements and constraints in terms of delay.

Figure 12 visually represents the power consumption of the wearable device during various time frames (rounds)for both Always-on (a) and Sync-ESP-NOW (b-f) at a random



(a) n=1, k=1 (b) n=1, k=1 (c) n=2, k=1

(d) n=2, k=3, multiple rounds (e) n=3, k=3, 1st hop (f) n=3, k=3, 2nd hop

Figure 12: Always-On (a) and Sync-ESP-NOW (b-f) power consumption with variable combinations of TTL and successive transmissions.

round or rounds using different TTL values and successive transmissions. The difference in power consumption is clearly demonstrated. In Sync-ESP-NOW, the device wakes up approximately 90ms before receiving the beacon, while some of this time is dedicated to boot up and the rest to the guard time to compensate for any clock drift (mainly for the RTOS scheduler). This means that the round duration is longer compared to 'Always-on' but the device is in sleep mode for the rest of the time, thus, spending the minimum possible energy. The beacon waiting time of Sync-ESP-NOW is automatically adjusted in multi-hop scenarios (see figures (d, f)).

5. Conclusion & Future Work

In conclusion, this paper introduced a portable and cost-effective signaling system designed to enhance safety by notifying workers on the rails about approaching trains in remote areas lacking permanent network infrastructure. The system leverages radio technologies and custom communication protocols to facilitate reliable and efficient communication. Initial experiments demonstrated promising results in terms of reliability, minimal packet losses, and energy efficiency for the wearable device.

Future research endeavors will focus on expanding the system's capabilities through extensive experiments conducted over longer distances and various terrain surfaces apart from the light obstacles type of terrain. We will also explore the possibility of using tethered drones to expand the coverage range of the system in hilly or forestry areas.

Acknowledgement

This publication has emanated from research conducted with the financial support of Samruk-Kazyna JSC for the project "Development of a warning system for repair crews about the approach of a train" grant No. 280921SK0201 implemented by Nazarbayev University for National Railways company JSC.

References

[1] B. Sahin, H. Yilmaz, Y. Ust, A. F. Guneri, B. Gulsun, An approach for analysing transportation costs and a case study, European Journal of Operational Research 193 (1) (2009) 1–11.

[2] C. Ngamkhanong, S. Kaewunruen, B. J. A. Costa, State-of-the-art review of railway track resilience monitoring, Infrastructures 3 (1) (2018) 3.

[3] A. Eduard, D. Urazayev, A. Sabyrbek, Y. Magzym, D. Zorbas, Infrastructure-Less Long-Range Train-Arrival Notification System, in: 2023 IEEE Symposium on Computers and Communications (ISCC), IEEE, 2023, pp. 956–962.

[4] G. R. Leone, M. Magrini, D. Moroni, G. Pieri, O. Salvetti, M. Tampucci, A smart device for monitoring railway tracks in remote areas, in: 2016 International Workshop on Computational Intelligence for Multimedia Understanding (IWCIM), 2016, pp. 1–5.

[5] S. Oh, S. Park, C. Lee, A platform surveillance monitoring system using image processing for passenger safety in railway station, in: 2007 International Conference on Control, Automation and Systems, IEEE, 2007, pp. 394–398.

[6] G. Alessandretti, A. Broggi, P. Cerri, Vehicle and guard rail detection using radar and vision data fusion, IEEE transactions on intelligent transportation systems 8 (1) (2007) 95–105.

[7] H. Mukojima, D. Deguchi, Y. Kawanishi, I. Ide, H. Murase, M. Ukai, N. Nagamine, R. Nakasone, Moving camera background-subtraction for obstacle detection on railway tracks, in: 2016 IEEE international conference on image processing (ICIP), IEEE, 2016, pp. 3967–3971.

[8] P. Sikora, L. Malina, M. Kiac, Z. Martinasek, K. Riha, J. Prinosil, L. Jirik, G. Srivastava, Artificial intelligence-based

surveillance system for railway crossing traffic, IEEE Sensors Journal 21 (14) (2020) 15515–15526.

[9] M. T. Baysari, A. S. McIntosh, J. R. Wilson, Understanding the human factors contribution to railway accidents and incidents in Australia, Accident Analysis & Prevention 40 (5) (2008) 1750–1757.

[10] W. Zhangyu, Y. Guizhen, W. Xinkai, L. Haoran, L. Da, A camera and LiDAR data fusion method for railway object detection, IEEE Sensors Journal 21 (12) (2021) 13442–13454.

[11] K. Sato, S. Ishida, J. Kajimura, M. Uchino, S. Tagashira, A. Fukuda, Initial evaluation of acoustic train detection system, in: Proc. ITS Asia–Pacific Forum Fukuoka, 2018, pp. 1–12.

[12] S. Ishida, K. Mimura, S. Liu, S. Tagashira, A. Fukuda, Design of simple vehicle counter using sidewalk microphones, in: Proc. ITS EU Congress. EU-TP0042, 2016, pp. 1–10.

[13] V. Dumont, V. R. Tribaldos, J. Ajo-Franklin, K. Wu, Deep learning for surface wave identification in distributed acoustic sensing data, in: 2020 IEEE international conference on big data, IEEE, 2020, pp. 1293–1300.

[14] D. Milne, A. Masoudi, E. Ferro, G. Watson, L. Le Pen, An analysis of railway track behaviour based on distributed optical fibre acoustic sensing, Mechanical Systems and Signal Processing 142 (2020) 106769.

[15] B. A. Khivsara, P. Gawande, M. Dhanwate, K. Sonawane, T. Chaudhari, IOT Based Railway Disaster Management System, in: 2018 Second International Conference on Computing Methodologies and Communication (ICCMC), 2018, pp. 680–685.

[16] P. L. Shrestha, M. Hempel, J. Santos, H. Sharif, A Multi-Sensor Approach for Early Detection and Notification of Approaching Trains, in: ASME/IEEE Joint Rail Conference, Vol. 45356, American Society of Mechanical Engineers, 2014, p. V001T06A003.

[17] I. Jain, S. Malik, S. Agrawal, Automatic Railway Barrier System, Railway Tracking and Collision Avoidance using IOT, International Journal of Computer Applications 975 (2017) 8887.

[18] H. J. Fu, Railway automatic safety protection system based on GPS, in: MATEC Web of Conferences, Vol. 44, EDP Sciences, 2016, p. 01061.

[19] O. DIENG, C. PHAM, O. THIARE, Outdoor Localization and Distance Estimation Based on Dynamic RSSI Measurements in LoRa Networks: Application to Cattle Rustling Prevention, in: 2019 International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob), 2019, pp. 1–6.

[20] M. Rademacher, H. Linka, T. Horstmann, M. Henze, Path Loss in Urban LoRa Networks: A Large-Scale Measurement Study, in: 2021 IEEE 94th Vehicular Technology Conference (VTC2021-Fall), 2021, pp. 1–6.

[21] L. Liu, Y. Yao, Z. Cao, M. Zhang, DeepLoRa: Learning Accurate Path Loss Model for Long Distance Links in LPWAN, in: IEEE Conference on Computer Communications (INFOCOM), 2021, pp. 1–10.

[22] M. Radeta, M. Ribeiro, D. Vasconcelos, H. Noronha, N. J. Nunes, LoRaquatica: Studying Range and Location Estimation using LoRa and IoT in Aquatic Sensing, in: 2020 IEEE International Conference on Pervasive Computing and Communications Workshops (PerCom Workshops), 2020, pp. 1–6.

[23] S. Demetri, M. Zúñiga, G. P. Picco, F. Kuipers, L. Bruzzone, T. Telkamp, Automated Estimation of Link Quality for LoRa: A Remote Sensing Approach, in: 2019 18th ACM/IEEE International Conference on Information Processing in Sensor Networks (IPSN), 2019, pp. 145–156.

[24] K.-H. Lam, C.-C. Cheung, W.-C. Lee, LoRa-based localization systems for noisy outdoor environment, in: 2017 IEEE 13th International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob), 2017, pp. 278–284.

[25] M. Anjum, M. A. Khan, S. A. Hassan, A. Mahmood, H. K. Qureshi, M. Gidlund, RSSI Fingerprinting-Based Localization Using Machine Learning in LoRa Networks, IEEE Internet of Things Magazine 3 (4) (2020) 53–59.

[26] K. Z. Islam, D. Murray, D. Diepeveen, M. G. K. Jones, F. Sohel, Machine learning-based LoRa localisation using multiple received signal features, IET Wireless Sensor Systems 13 (4) (2023) 133–150.

[27] F. Carrino, A. Janka, O. Abou Khaled, E. Mugellini, LoRaLoc: Machine Learning-Based Fingerprinting for Outdoor Geolocation using LoRa, in: 2019 6th Swiss Conference on Data Science (SDS), 2019, pp. 82–86.

[28] M. C. Bor, U. Roedig, T. Voigt, J. M. Alonso, Do LoRa Low-Power Wide-Area Networks Scale?, in: Proceedings of the 19th ACM International Conference on Modeling, Analysis and Simulation of Wireless and Mobile Systems, MSWiM '16, ACM, 2016, pp. 59–67.

[29] H. Linka, M. Rademacher, O. G. Aliu, K. Jonas, Path loss models for low-power wide-area networks: Experimental results using LoRa, VDE Verlag, 2018.

[30] R. Anzum, M. H. Habaebi, M. R. Islam, G. P. Hakim, M. U. Khandaker, H. Osman, S. Alamri, E. AbdElrahim, A multi-wall path-loss prediction model using 433 MHz LoRa-WAN frequency to characterize foliage's influence in a Malaysian palm oil plantation environment, Sensors 22 (14) (2022) 5397.

[31] A. De Myttenaere, B. Golden, B. Le Grand, F. Rossi, Mean absolute percentage error for regression models, Neurocomputing 192 (2016) 38–48.

[32] J. Jin, J. Yan, K. Chen, Transfer learning with quantile regression (2023). arXiv:2212.06693.

[33] A. Eduard, D. Urazayev, Y. Magzym, D. Zorbas, Demo: Implementation of a Train-Arrival Notification System, in: IEEE Symposium on Computers and Communications (ISCC), IEEE, 2023, pp. 1–3.

[34] Y. Magzym, A. Eduard, D. Urazayev, X. Fafoutis, D. Zorbas, Synchronized ESP-NOW for Improved Energy Efficiency, in: IEEE International Black Sea Conference on Communications and Networking, IEEE, 2023, pp. 1–6.

[35] T. N. Hoang, S.-T. Van, B. Nguyen, ESP-NOW based decentralized low cost voice communication systems for buildings, in: 2019 International Symposium on Electrical and Electronics Engineering (ISEE), IEEE, 2019, pp. 108–112.

[36] H. A. Nguyen, L. V. Nguyen, Q. P. Ha, IoT-enabled dependable co-located low-cost sensing for construction site monitoring, in: 37th International Symposium on Automation and Robotics in Construction, International Association for Automation and Robotics in Construction (IAARC), 2020.

[37] L. Wawrla, O. Maghazei, T. Netland, Applications of drones in warehouse operations, Whitepaper. ETH Zurich, D-MTEC (2019) 212.

[38] S. H. Alsamhi, F. A. Almalki, O. Ma, M. S. Ansari, M. C. Angelides, Performance optimization of tethered balloon technology for public safety and emergency communications, Telecommunication Systems 75 (2020) 235–244.