Data and Metadata. 2025; 4:829
doi: 10.56294/dm2025829

ORIGINAL

Route Optimization for Urban Waste Collection via LoRa Communication and Deep Q-Network in a Controlled Simulation

Optimización de Rutas para la Recolección de Residuos Urbanos mediante Comunicación LoRa y Deep Q-Network en una Simulación Controlada

Gabriela Caragulla-Briceño[1], Francisco Ger-Ortega[1], Fabián Cuzme-Rodríguez[1], Luis Suárez-Zambrano[1], Alejandra Pinto-Erazo[1], Henry Farinango-Endara[1]

[1]Universidad Técnica del Norte, Carrera de Telecomunicaciones. Ibarra, Ecuador.

Cite as: Caragulla-Briceño G, Ger-Ortega F, Cuzme-Rodríguez F, Suárez-Zambrano L, Pinto-Erazo A, Farinango-Endara H. Route Optimization for Urban Waste Collection via LoRa Communication and Deep Q-Network in a Controlled Simulation. Data and Metadata. 2025; 4:829. https://doi.org/10.56294/dm2025829

Submitted: 11-07-2025 **Revised:** 08-09-2025 **Accepted:** 03-12-2025 **Published:** 04-12-2025

Editor: Dr. Adrián Alejandro Vitón Castillo

Corresponding Author: Fabián Cuzme-Rodríguez

ABSTRACT

Traditional urban solid waste management in Latin American cities operates using fixed routes, without considering the dynamic state of waste containers. This rigid approach leads to operational inefficiencies, excessive fuel consumption, and a significant environmental impact. A distributed architecture was designed that integrates HC-SR04 sensors (fill level) and MQ-135 sensors (air quality), connected to Heltec WiFi LoRa 32 V3 nodes. The captured data is transmitted via LoRa technology to a central gateway and stored in real-time on Firebase. These data were used to train a Deep Q-Network (DQN) model, developed in PyTorch using OpenAI Gym, with an input of 30 parameters (15 containers × 2 variables) and 15 possible actions. Training was performed over 1000 epochs with a learning rate of 0,0005 and a discount factor γ = 0,99. The model achieved a stable decision policy to dynamically prioritize critical collection points. Compared to static routes, there was a 16,4 % reduction in distance traveled, 16,3 % in operational time, and 16,4 % in fuel consumption. Route planning was complemented by the Dijkstra algorithm and visualized in a geospatial interface using the Google Maps API. The system was implemented as a Flask API integrated with a hybrid mobile application, allowing real-time visualization of optimized routes and container status. This intelligent and scalable solution reduces resource usage, improves urban sustainability, and is well-suited for deployment in smart cities.

Keywords: Routing Algorithms; Deep Reinforcement Learning (DQN); LoRa Technology; Smart Cities; Internet of Things (IoT).

RESUMEN

La gestión tradicional de residuos sólidos urbanos en ciudades latinoamericanas opera bajo rutas fijas, sin considerar el estado dinámico de los contenedores. Este enfoque rígido genera ineficiencias operativas, consumo excesivo de combustible y un impacto ambiental significativo. Se diseñó una arquitectura distribuida que integra sensores HC-SR04 (nivel de llenado) y MQ-135 (calidad del aire) conectados a nodos Heltec WiFi LoRa 32 V3. La información capturada se transmite mediante tecnología LoRa a un gateway central, siendo almacenada en tiempo real en Firebase. Los datos fueron utilizados para entrenar un modelo Deep Q-Network (DQN), desarrollado en PyTorch sobre OpenAI Gym, con una entrada de 30 parámetros (15 contenedores × 2 variables) y 15 posibles acciones. El entrenamiento se realizó durante 1000 épocas con una tasa de aprendizaje de 0,0005 y un factor de descuento γ = 0,99. El modelo logró una política de decisión estable para priorizar dinámicamente los puntos críticos de recolección.

En comparación con rutas estáticas, se evidenció una reducción del 16,4 % en distancia recorrida, 16,3 % en tiempo operativo y 16,4 % en consumo de combustible. La planificación de rutas fue complementada mediante el algoritmo de Dijkstra y visualizada en una interfaz geoespacial sobre Google Maps API. El sistema fue implementado como una API Flask integrada con una aplicación móvil híbrida, lo que permite visualizar en tiempo real las rutas optimizadas y el estado de los contenedores. Esta solución inteligente y escalable reduce el uso de recursos, mejora la sostenibilidad urbana y resulta adecuada para su despliegue en ciudades inteligentes.

Palabras clave: Algoritmos de Enrutamiento; Aprendizaje por Refuerzo Profundo (DQN); Tecnología LoRa; Ciudades Inteligentes; Internet de las Cosas (IoT).

INTRODUCTION

Urbanization has complicated waste management, and traditional fixed-route methods lead to inefficiencies. A dynamic routing system is proposed, using real-time analytics and intelligent algorithms to optimize waste collection in evolving urban environments.(1) Waste management represents a growing challenge due to population growth and the increasing volume of waste generated. Traditional systems, with non-optimized routes and inefficient planning, lead to high operational costs and environmental degradation. In response, intelligent approaches based on IoT and machine learning have been developed to predict fill levels and anticipate future waste generation, enhancing efficiency in smart cities.(2)

Recent assessments show a sustained rise in municipal solid waste (MSW), with global projections increasing from ~2,1 billion tonnes in 2023 to ~3,8 billion tonnes by 2050, intensifying environmental and fiscal burdens if business-as-usual practices persist. These costs span GHG emissions, air-quality impacts, and biodiversity losses, reinforcing the urgency for data-driven, adaptive collection systems in urban settings.(3)

In response to this issue, the integration of intelligent solutions based on emerging technologies such as the Internet of Things (IoT), low-power communication networks like LoRa (Long Range), and deep learning algorithms is proposed. These technologies enable distributed systems capable of real-time monitoring of container status and dynamic optimization of waste collection routes. Within this framework, artificial intelligence (AI) offers a highly promising complementary approach, as recent studies indicate it presents new opportunities to address the increasing complexity of waste management systems, including collection, sorting, recycling, and monitoring.(4) However, to maximize its impact, it is essential to address challenges related to data quality, privacy, operational costs, and the ethical frameworks that govern its implementation.

Over the last five years, smart-waste deployments have increasingly adopted IoT with long-range, low-power networks (LoRa/LoRaWAN) for city-scale bin monitoring, and deep reinforcement learning to cope with stochastic, time-varying routing decisions. Field trials confirm LoRaWAN's suitability for low-throughput sensing over large urban areas, while deep RL, particularly DQN, improves decision policies under partial observability compared to static graph search alone. Unlike Dijkstra or A* (which assume static, fully known graphs) and population-based heuristics (e.g., genetic algorithms), DQN learns a state-value policy directly from interaction, capturing delayed rewards and evolving constraints without prescribing fixed cost functions. This motivates our choice of LoRa-based sensing coupled with a DQN policy for adaptive container prioritization.(5,6,7,8,9,10)

This project proposes an intelligent architecture combining ultrasonic sensors, LoRa communication, and a Deep Q-Network (DQN) model trained in controlled simulations to select and prioritize critical containers. This technological integration aims to enhance operational efficiency, reduce environmental impact, and advance toward a more sustainable city model.

In Ibarra, solid waste collection is currently managed through fixed routes defined by the local government, with separate schedules for urban and rural zones. Collection is performed by conventional collection vehicles that follow predetermined paths, regardless of whether containers are full or nearly empty. While certain areas have seen improvements like the introduction of waste bins, the system remains largely static and unresponsive to actual waste generation patterns. This often results in inefficient routing, unnecessary fuel consumption, and operational strain on municipal services.(11)

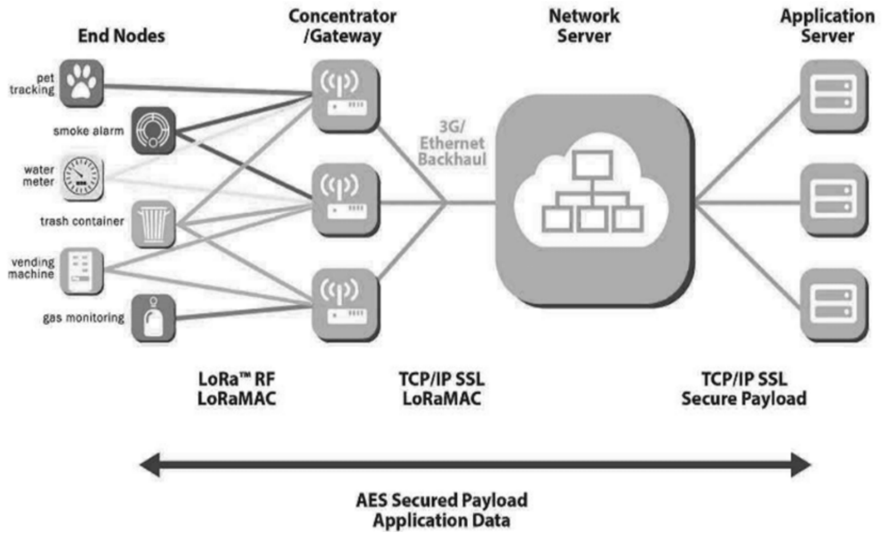
A solution for urban waste management using LoRaWAN sensors can be assessed by evaluating its performance in real-world scenarios. The proposed approach validates the technical feasibility of an LPWAN network to monitor fill levels and enhance municipal operational efficiency.(12) In another study, a waste collection system based on Deep Reinforcement Learning (DRL) is proposed, utilizing autonomous vehicles and smart containers connected via IoT. Using Deep Q-Networks, routes are optimized based on traffic conditions and available energy, resulting in significant improvements in operational efficiency and energy consumption.(13)

Building on this approach, the present proposal integrates Deep Q-Networks (DQN) as the core of the route optimization system, leveraging data generated by ultrasonic sensors installed in waste containers and transmitted via LoRa technology. Unlike other solutions, this system emphasizes hardware simplicity and scalability, enabling dynamic route planning that minimizes unnecessary travel and enhances energy efficiency.

LoRa (Long Range)

LoRa (Long Range) is a wireless communication technology designed to transmit data over long distances while consuming minimal energy, making it an ideal choice for Internet of Things (IoT) applications.[14](1) As part of Low Power Wide Area Networks (LPWAN), LoRa enables the connection of distributed devices across large areas without the need for costly network infrastructure or high maintenance requirements.

Its structure is based on end devices (sensors or nodes), gateways (communication bridges), and network servers that manage data traffic. This modular architecture enables the development of scalable private networks, as highlighted in recent studies describing the design and implementation of customized gateways and LoRa servers segmented into independent modules for greater operational flexibility. Due to its efficiency, low cost, and ease of deployment, LoRa is particularly well-suited for monitoring complex urban environments, such as the smart waste collection system proposed in this work.



Source: Croce et al.[15]

Figure 1. LoRa Network Architecture for Long-Range IoT Communication

The architecture consists of end devices that transmit data via LoRa to gateways, which then relay the information to the network server through an IP connection. The server manages and centralizes communication, enabling an efficient and low-power IoT network.[16]

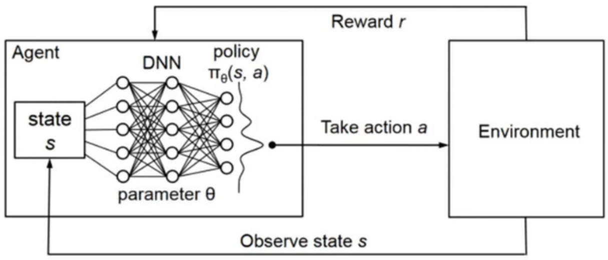
It operates in unlicensed frequency bands, such as 868 MHz in Europe and 915 MHz in the Americas. Its range can exceed 10 km in rural areas and between 2 to 5 km in urban environments. LoRa uses Chirp Spread Spectrum modulation, which provides high interference tolerance and ensures robust communication even under challenging conditions. It supports low data transmission rates (from 0,3 kbps to 50 kbps), making it suitable for sending small data packets.[17] LoRa is highly energy-efficient, allowing devices to operate for years on standard batteries. The technology supports bidirectional communication, data encryption, and scalability to thousands of nodes within a single network.[18]

This technology was selected for its balance between long-range coverage, low energy consumption, and low implementation cost, making it ideal for urban environments like Ibarra. Unlike GSM or Wi-Fi networks, LoRa enables the deployment of a dedicated infrastructure without recurring fees, ensuring stable connectivity even in areas with limited coverage.[19]

Deep Q-Network (DQN)

Deep Q-Network (DQN) is a deep reinforcement learning approach that combines neural function approximation with Q-learning to select the next container to service under dynamic conditions.[20] Its objective is to train an agent to make optimal decisions in dynamic environments by maximizing cumulative rewards through continuous interaction with the environment.

As shown in figure 2, DQN uses a neural network to estimate action values (Q-values), allowing it to select the most appropriate action based on the current state of the environment. The agent learns through experience, receiving rewards for actions that move it closer to a goal and penalties for undesirable decisions, such as unnecessary movements or collisions.



Source Jayasekara[21]
Figure 2. Structure of a Deep Q-Network (DQN) in a reinforcement learning environment

An agent observes the state of the environment, processes it through a deep neural network (DNN), and takes an action. It then receives a reward and updates its decision-making policy accordingly.[22]

This approach has been successfully applied to complex tasks such as autonomous robot navigation in dynamic environments. It is first trained in simulations and then implemented in real-world scenarios, demonstrating strong adaptability and real-time optimization capabilities. In the proposed project, this technique is used to dynamically select the containers that need to be collected, as shown in figure 3, taking into account factors such as fill level and route efficiency.

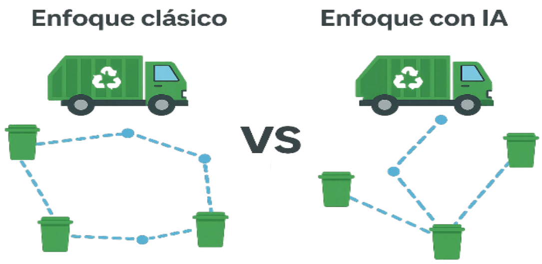


Figure 3. Comparison between fixed collection route and route optimized through intelligent algorithms

The figure compares two waste collection approaches: on the left, a fixed route that collects all containers regardless of their fill level; on the right, an optimized route that only visits containers that actually require service.

METHOD

System architecture

The system architecture is composed of four functional layers. In the device layer, ultrasonic sensors installed in containers detect fill levels and transmit the data via LoRa nodes. In the network layer, these sensor nodes send the information to a gateway using LoRa technology, which then relays it via Wi-Fi to a central database. In the application support and services layer, the data is processed and used to train a Deep Q-Network (DQN) model that learns to optimize collection routes. In the application layer, the trained model generates efficient routes that are displayed through an interface for the operator in charge.

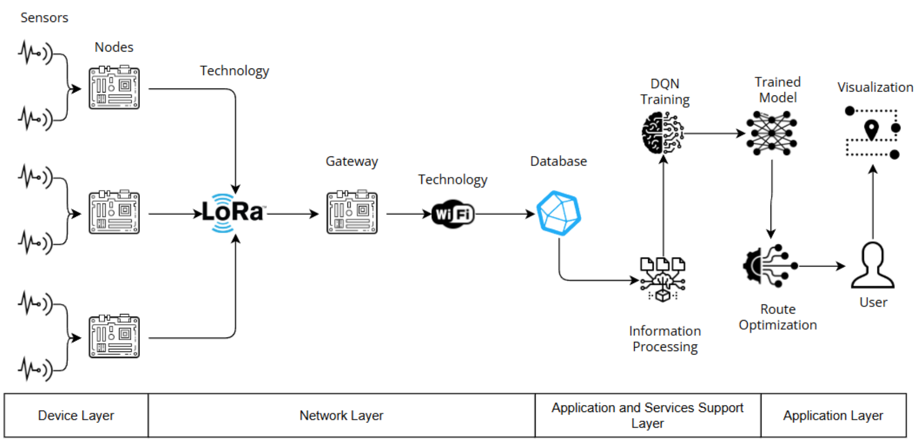


Figure 4. Proposed System Architecture

This modular architecture implements a distributed data acquisition and transmission infrastructure, where ultrasonic sensors connected to LoRa nodes capture the fill levels of waste containers and transmit the information via LPWAN links. The data is centralized through a gateway that uses Wi-Fi connectivity to forward it to a central database.

Subsequently, a processing pipeline is applied to structure and normalize the data, preparing it for input into a deep reinforcement learning model (Deep Q-Network). Trained in a simulated environment, this model generates optimal actions for dynamic route planning, and its results are integrated into a graphical interface for operational visualization.

Processes

a) Data acquisition and preprocessing. Each sensor node samples fill level (HC-SR04) and air-quality proxy (MQ-135) and transmits compact payloads via LoRa to the gateway, which relays them over Wi-Fi to Firebase. The backend (Flask) retrieves raw records, applies timestamp alignment, outlier filtering (3σ rule), min-max normalization to [0,1] per feature, and persists curated batches for training.

b) Training pipeline (DQN). The DQN agent learns a policy that prioritizes the next container to service from a state that stacks normalized *fill and air* features for all containers. To ensure stable learning in this dynamic, partially observable setting, we use the standard DQN recipe: a target network (stabilizes updates), experience replay (decorrelates samples), and a decaying ε-greedy exploration policy (balances exploration and exploitation). The resulting policy adapts to changing conditions and produces a value-aware prioritization rather than relying on fixed, static costs.

c) Inference and route generation. At runtime, the mobile client sends the current state to the Flask API. The policy returns the next container; selected containers are masked to avoid repetition and to produce a prioritized list. This ordered set feeds a directed road graph (street directionality enforced); the shortest path is computed with Dijkstra and rendered on a web map (Google Maps API) for the collection vehicle operator.

The diagram in figure 5 outlines the complete system flow, from data collection to the visualization of optimized routes.

The stored data is retrieved and preprocessed using Pandas and Numpy, preparing it for model training within a simulated environment using OpenAI Gym and PyTorch. In this setup, a Deep Q-Network (DQN) algorithm learns to select optimal routes based on the status of the containers.

Once trained, the model is deployed as an API via Flask. Users can access a web interface to request the optimal route, which is then generated and displayed using Google Maps API. This system integrates IoT technologies, deep learning, and web services to provide a dynamic and efficient waste collection solution.

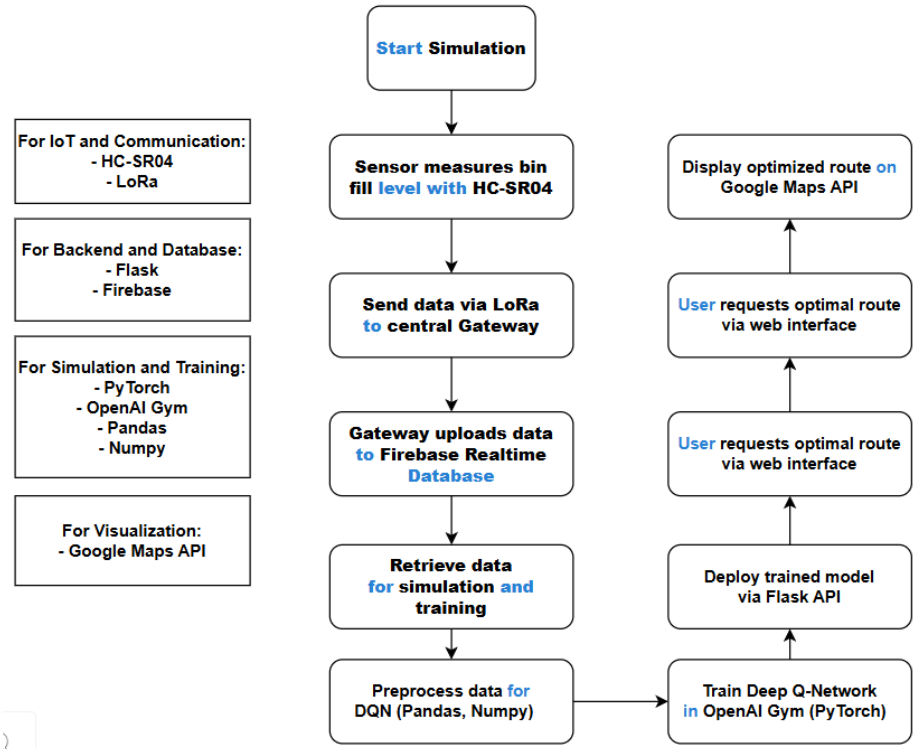
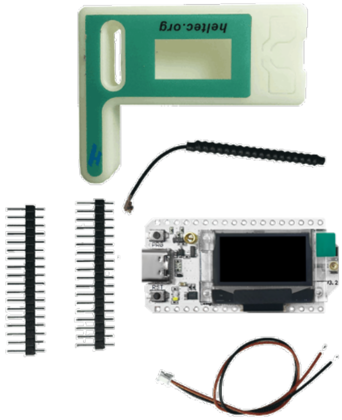


Figure 5. System Flow Diagram

https://doi.org/10.56294/dm2025829 ISSN: 2953-4917

Development board

The Heltec WiFi LoRa 32 V3 (HTIT-WB32LA) is a low-power IoT development board that integrates an ESP32-S3FN8 microcontroller (dual-core, 240 MHz), Wi-Fi and Bluetooth connectivity, and an SX1262 LoRa module for long-range communication in the 863–928 MHz frequency range. It features a 0,96” OLED display (128×64), a USB Type-C port, a LoRa antenna with IPEX interface, and an integrated 3,7 V battery management system with overcharge protection.[23]



Source: Heltec Automation[23]
Figure 6. Components of the Heltec WiFi LoRa 32 V3 development module

Its design is optimized for low power consumption, achieving a receiver sensitivity of up to –134 dBm and a maximum transmission power of 21 dBm, making it an ideal solution for LPWAN networks in urban environments.

System design

The sensor node connection is powered by a 3,7 V, 1000 mAh battery, suitable for low-power applications. Two sensors are connected to the board: the HC-SR04, which measures distance to estimate the container’s fill level, and the MQ-135, which detects the concentration of air pollutants (figure 7). Both sensors share power and ground lines and send signals to the GPIO pins of the Heltec board. LoRa communication is used to wirelessly transmit the collected data to the gateway for further processing.

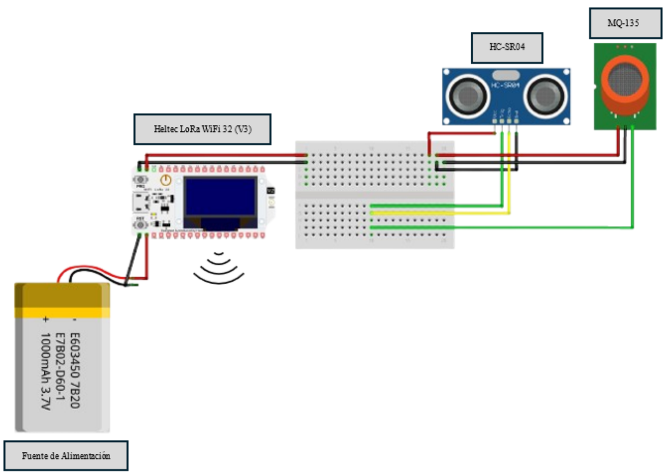


Figure 7. Connection diagram of the sensor node with Heltec WiFi LoRa 32 V3, HC-SR04, and MQ-135

The presented design enables efficient integration of both analog and digital sensors into a single autonomous node. The Heltec board simultaneously manages data acquisition and LoRa transmission, while the rechargeable battery power supply ensures operation in remote locations without electrical infrastructure. The modular assembly design facilitates easy replication and field maintenance.

The system is designed to be deployed in solid waste containers, as illustrated in figure 8, located in urban environments specifically in strategic areas of Ibarra, Ecuador. These locations include residential, commercial, and high-traffic zones where waste accumulation tends to be more frequent. The goal is to monitor container fill levels in real-time and optimize both the frequency and routing of waste collection, thereby improving the operational efficiency of municipal services and reducing the associated environmental impact.

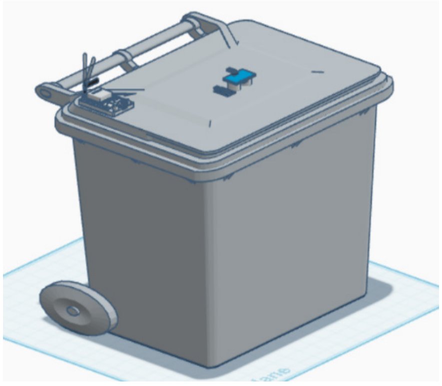


Figure 8. 3D Model of the Smart Container with Integrated Sensors

In the proposed model, the sensing and transmission system is integrated directly into the container's structure. A compact module mounted on the lid houses the Heltec board, sensors, and LoRa antenna. This configuration allows for a functional and aesthetically clean installation, protecting the electronic components from environmental factors and facilitating maintenance. The design is fully autonomous and adaptable to standard urban waste containers.

DEVELOPMENT

To validate the system's behavior prior to physical deployment, a simulation was developed to emulate the operation of 15 sensor nodes attached to geolocated containers in Ibarra. Each node produces normalized fill and air-quality signals in accelerated time to emulate field dynamics.

Data are produced in accelerated time and ingested through the same Firebase → Flask path used at runtime, mirroring the continuous flow from physical nodes. The simulation enables the adjustment and training of the route optimization model without requiring active hardware, thus facilitating preliminary system validation.

```python
# === Coordenadas reales de los 15 basureros ===
contenedores = {
    "C1": {"lat": 0.3540853135771185, "lng": -78.12040183409611},
    "C2": {"lat": 0.3548255891063949, "lng": -78.1193289505281},
    "C3": {"lat": 0.35464320239261643, "lng": -78.11823460930029},
    "C4": {"lat": 0.35552295003772005, "lng": -78.1186208473807},
    "C5": {"lat": 0.35410677086066206, "lng": -78.1187495934075},
    "C6": {"lat": 0.3525189333443036, "lng": -78.11898562778997},
    "C7": {"lat": 0.3514031554680615, "lng": -78.11918947566575},
    "C8": {"lat": 0.35122193371184723, "lng": -78.12048130315775},
    "C9": {"lat": 0.3532483427290123, "lng": -78.12112086341519},
    "C10": {"lat": 0.35308595270106047, "lng": -78.12022770131776},
    "C11": {"lat": 0.3527792159738607, "lng": -78.11845039896227},
    "C12": {"lat": 0.35206650412751456, "lng": -78.1196503035982},
    "C13": {"lat":  0.3522288941732521, "lng": -78.12105771053962},
    "C14": {"lat": 0.3528062809796071, "lng": -78.1207780335192},
    "C15": {"lat": 0.3515883556431086, "lng": -78.12098553582467}
}

# === Estado inicial de cada contenedor ===
estado_contenedores = {
    cid: {"nivel": random.uniform(0, 20), "aire": random.uniform(100, 150)}
    for cid in contenedores
}

# === Historial de datos (se guarda en CSV al final) ===
historial = []

# Total de ciclos de simulación (ej: 5400 = 3 horas con 2 seg por ciclo)
total_ciclos = 500
ciclo = 0
```

Figure 9. Code snippet for simulating container coordinates and initial conditions

This simulation defines 15 geolocated containers and generates virtual fill and air-quality data in accelerated cycles; figure 9 shows a representative code snippet.

Previously selected containers are masked, ensuring that the prioritized list produced at inference is consistent with the policy described in Methods (Processes).

Model Training

Training follows the DQN procedure summarized in Methods. The agent iteratively observes the city state and selects the next container, receiving feedback that promotes prioritizing critical bins. Learning curves are monitored and the best checkpoint is deployed through the Flask API. Using simulated data representing the fill levels and air quality at each collection point, the agent makes decisions in each cycle, receiving rewards or penalties based on the accuracy of its choices. This iterative process allows the model to refine its policy to maximize operational efficiency, optimizing routes according to real-world environmental conditions.

The reward promotes servicing high-priority containers (high fill, poor air-quality proxy) and discourages low-impact selections. This simple, normalized signal is sufficient to bias learning toward operationally meaningful actions, consistent with the city's objectives.

The agent was trained for 1000 epochs with a learning rate of 0,0005 and a discount factor γ = 0,99. The input dimensionality is 2×N (fill and air per container; N = 15), and the output layer has N discrete units, each corresponding to one container selection action. These settings were kept fixed across runs to ensure comparability and consistency with the Methods description.

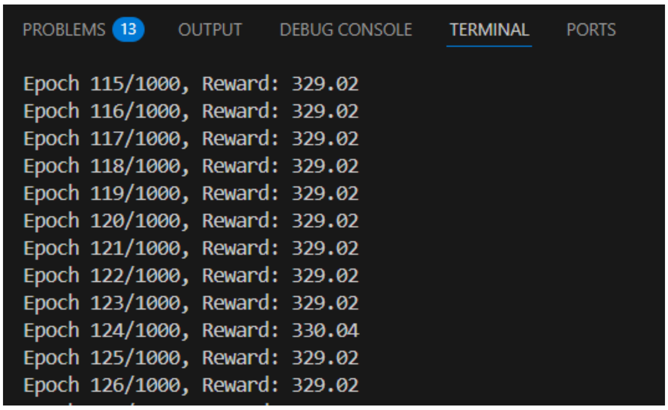


Figure 10. Training process of the DQN model

The result shown in figure 10 corresponds to the training process of the DQN model, where a stable total reward close to 329,02 is consistently observed over several consecutive epochs, with slight variations such as in epoch 124. This behavior indicates that the agent has reached a stable and efficient policy for decision-making within the simulated environment. The consistency in reward suggests that the model has properly converged, optimizing its performance in selecting containers that require priority collection.

Selection Method

The selection method implemented in the system relies on the policy learned by the Deep Q-Network (DQN) model, which determines the container to service in each cycle. Based on the current system state—defined by the fill levels and air quality of all containers the agent evaluates possible actions and selects the one with the highest Q-value. This approach enables dynamic prioritization of the most critical collection points, optimizing resource utilization and reducing unnecessary routes.

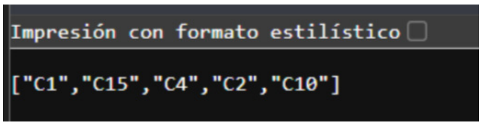


Figure 11. System output after applying the trained model

The system defines a priority order for waste collection. In this case, containers C1, C15, C4, C2, and C10 have been identified as the most critical based on their current status. This list reflects the decisions made by the DQN model agent, taking into account variables such as fill level and air quality. The model receives, in real-time, a structured array in JSON format from a client interface developed in Ionic. Each object in the array represents the state of a container, including key parameters such as the percentage of fill and air quality. As illustrated in figure 12, this information is sent to the backend via an HTTP request, where the server processes the data and transforms it into input vectors to feed the pre-trained Deep Q-Network (DQN) model.

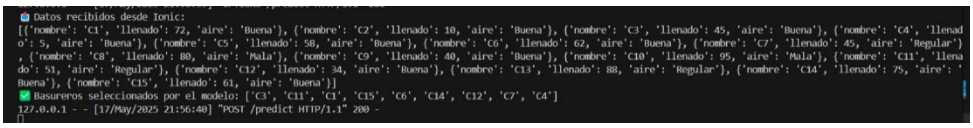


Figure 12. Real-time model execution receiving data from a mobile application developed in Ionic

Once processed, the model evaluates the complete state of the environment and generates an ordered sequence of actions representing the containers with the highest priority for service. In this case, the model

https://doi.org/10.56294/dm2025829   ISSN: 2953-4917

returns the prioritized containers [‘C3’, ‘C11’, ‘C1’, ‘C15’, ‘C6’, ‘C14’, ‘C12’, ‘C7’, ‘C4’], prioritizing those with high waste levels and/or poor air quality. This output optimizes the collection order and can later be visualized in a graphical environment such as the Google Maps API. An HTTP 200 response confirms the successful execution of the request and the prediction pipeline.

Route Calculation

Once the containers have been prioritized by the model, the Dijkstra algorithm is applied to determine the optimal collection route. This algorithm computes the shortest path between the selected nodes, minimizing the total travel distance. The road network used respects the actual directionality of streets, ensuring that the generated routes are viable for execution in urban environments.

The integration of the DQN model with the Dijkstra algorithm enables the system not only to identify which containers require service, but also to determine the most efficient order and path to reach them optimizing both operational time and fuel consumption.

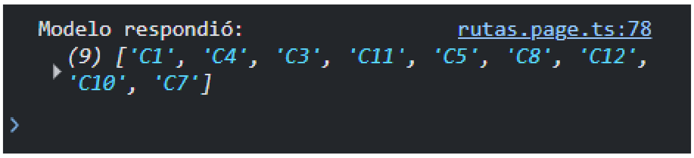


Figure 13. Model response with the prioritized list of containers for collection

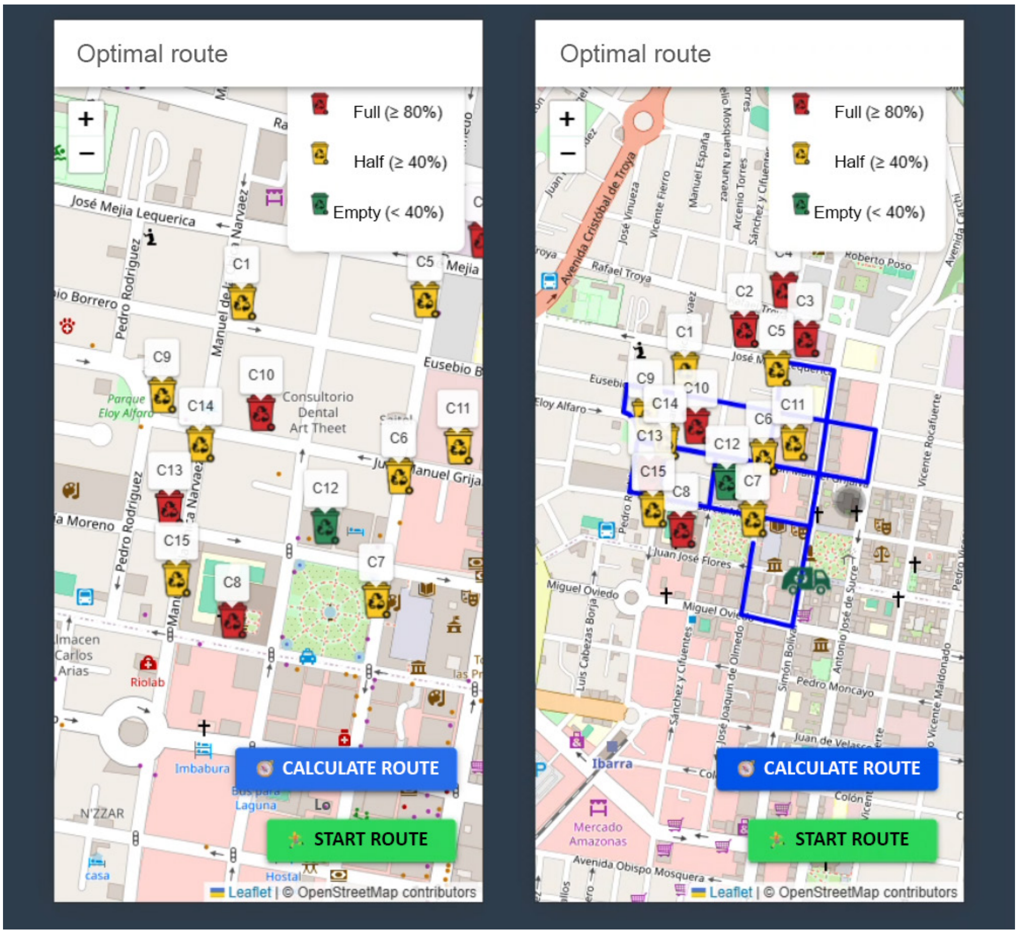


Figure 14. Interactive map in the app showing the optimal route and real-time location

The model has returned a list of nine containers selected as a priority for collection. This sequence has already been processed based on the individual status of each container and will subsequently be used as input for the route calculation algorithm.

Next, the application displays an interactive map on its interface, representing the spatial distribution of the containers along with their respective fill levels, classified using a color-coded scheme (red, yellow, and green). Once the model provides the prioritized containers, the system enables calculation and visualization of the optimal route using the path generated by the Dijkstra algorithm, while respecting actual street directions.

This route is overlaid directly on the map, giving the user a clear view of the order and path to follow. The interface also enables real-time tracking of the route, enhancing operational control over the waste collection tasks.

The route respects real-world urban road constraints and connects only the containers previously selected by the DQN model, optimizing both time and resource usage.

RESULTS

This section summarizes the learning behavior of the policy and contrasts operational indicators against a traditional fixed route. As shown in table 1, the AI-optimized approach achieves consistent improvements across the three key metrics, distance, time, and fuel consumption, providing a clear quantitative view of the gains attributable to the prioritization layer.

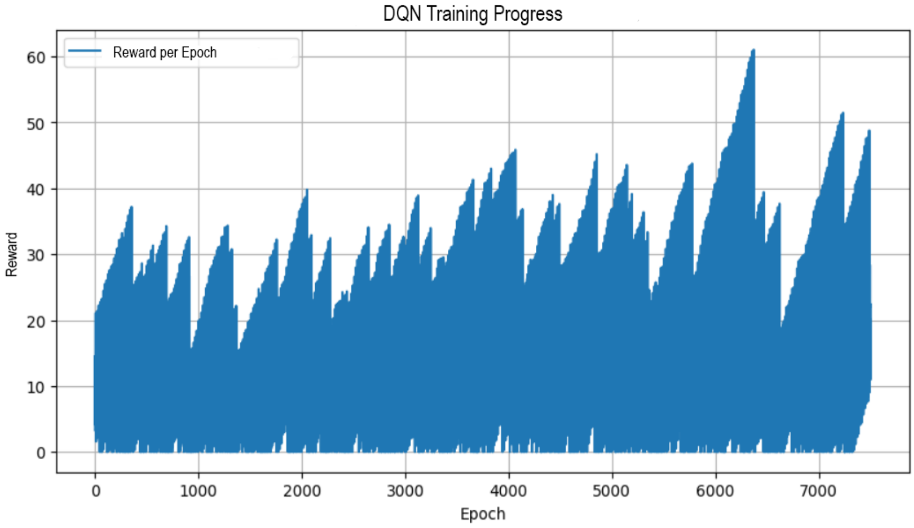


Figure 15. Evolution of the reward obtained over more than 7000 training epochs of the agent

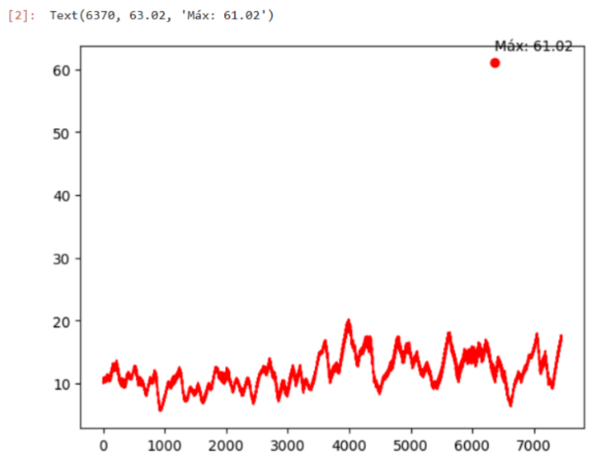


Figure 16. Reward evolution per epoch during the training of the DQN model

Figures 15-16 show a consistent upward trend in training rewards, with occasional fluctuations typical of the exploration-exploitation trade-off. After the mid-training phase, the agent stabilizes around higher reward levels and reaches a peak near epoch ~6,370, indicating a robust and generalizable policy for container prioritization under dynamic conditions.

Building on these learning dynamics, table 1 confirms consistent reductions across the three indicators, distance, time, and fuel, aligning with the expected effect of value-aware prioritization.

Figure 17 visually contrasts both approaches across distance, time, and fuel, reinforcing the uniform advantage of the AI-optimized route already quantified in table 1. These improvements clearly demonstrate the efficiency of the AI algorithm in selecting not only the most critical containers but also the most operationally optimal visiting sequence.

Table 1. Comparative results of route optimization using artificial intelligence

AI Route Optimization			
Segment	**Distance (km)**	**Time (h)**	**Consumption (L)**
C15 → C6	0,24525646	0,00817516	0,01962037
C6 → C11	0,06616749	0,00220558	0,00529339
C11 → C3	0,20750449	0,00691682	0,01660036
C3 → C4	0,10635578	0,00354519	0,00850846
C4 → C1	0,25411671	0,00847056	0,02032934
TOTAL	0,87939907	0,0293133	0,07035193
Traditional Fixed Route			
C3 → C4	0,10635578	0,00354519	0,00850846
C4 → C6	0,33464025	0,01115468	0,02671220
C6 → C11	0,06616744	0,00220558	0,00529339
C11 → C1	0,26085625	0,00869521	0,02086850
C1 → C15	0,28364219	0,00945474	0,02269138
TOTAL	1,05166191	0,0350554	0,08413295
Improvement (%)	~16,4 %	~16,3 %	~16,4 %

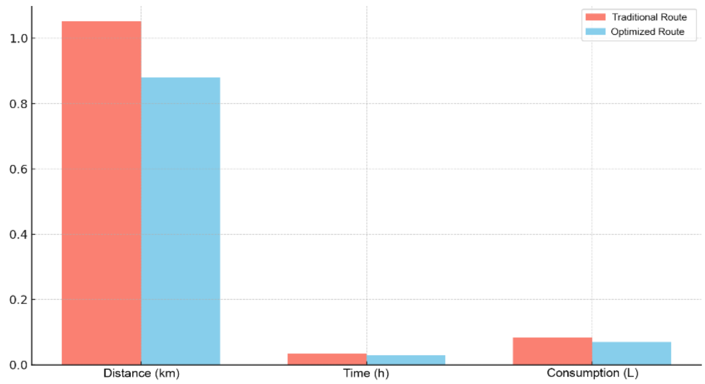


Figure 17. Route Comparison: Traditional vs AI

This type of visualization provides immediate insight into the positive impact of implementing an intelligent collection system, not only in terms of sustainability and energy efficiency but also in the optimization of logistical resources. This translates into significant benefits for urban waste management systems, reinforcing the value of AI-driven solutions for smart and sustainable city operations.

DISCUSSION

The implementation of the proposed system based on Deep Q-Networks (DQN) and LoRa communication led to a significant improvement in the dynamic planning of waste collection routes. The results were compared with previous research to validate the system's performance and scalability.

The observed gains stem from combining DQN-based prioritization with shortest-path routing. Unlike static graph search (e.g., Dijkstra/A*) or hand-crafted heuristics, the DQN policy learns from evolving states (fill and air-quality signals), adapting container selection under stochastic, time-varying conditions typical of urban operations. This adaptive layer complements the road-network optimizer rather than replacing it, explaining the sustained improvements in distance, time, and fuel.

Unlike the study by Cruz et al.[12], which focused solely on the technical feasibility of LoRaWAN networks for waste monitoring, the present work integrates real sensors, reinforcement learning model training, and route calculation within simulated scenarios featuring dynamic conditions. While Cruz and collaborators validated aspects like connectivity and coverage, this proposal extends the scope toward active route optimization, showing tangible impact across three key metrics: distance, time, and fuel consumption.

In comparison with Kavitha et al.[13], who implemented a DRL-based autonomous collection system, this work targets a realistic and deployable solution for Latin American municipalities. It leverages affordable hardware such as Heltec WiFi LoRa 32 V3 boards and low-cost sensors. Although both studies apply DQN, this research emphasizes a modular, distributed architecture and a mobile integration API, enabling immediate deployment in urban settings.

This comparative analysis highlights the system's practical advantages: not only does it achieve technical improvements, but it also ensures economic feasibility and real-world applicability—crucial factors for adoption in developing urban infrastructures.

Terminology was standardized throughout: sensor node denotes the embedded unit (Heltec + sensors), gateway the LoRa-to-IP bridge, and the municipal collection vehicle executing the computed route.

Robustness and Scalability

At the design level, the modular architecture of the system allows for the addition of more sensors or containers without modifying the core logic of the model. The integration of LoRa communication with Firebase enables scalability across cities with varying urban topologies, effectively overcoming the limitations of GSM or WiFi networks in areas with weak infrastructure.

The reward signal combined fill level and an air-quality proxy to increase sensitivity to operationally critical bins. This simple, normalized design produced more discriminative policies than distance-only objectives, consistently guiding the agent toward higher-impact actions.

Comparative Visualization

To complement the numerical analysis, a radar chart (Figure 18) was developed to visually synthesize the efficiency differences between the two approaches. It highlights three key metrics: total distance traveled, operation time, and fuel consumption. All values were normalized relative to the maximum value for each parameter to facilitate comparison.

As shown in the chart, the route optimized by artificial intelligence (AI) lies closer to the center in all dimensions, indicating lower resource usage and greater operational efficiency compared to the traditional fixed route. This visual representation reinforces the quantitative findings and clearly illustrates the practical advantage of AI-driven route planning.

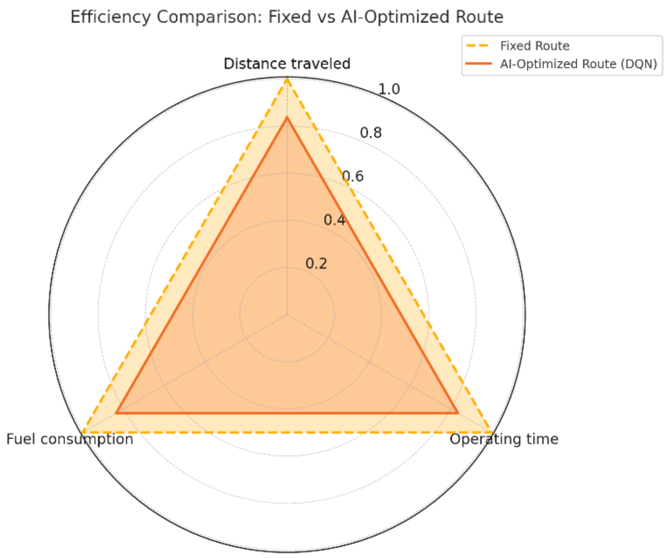


Figure 18. Radar plot comparing the efficiency of traditional fixed-route versus AI-optimized route using Deep Q-Networks (DQN). Lower values indicate higher efficiency across metrics

CONCLUSIONS

The implementation of the Deep Q-Network (DQN) algorithm enabled the calculation of waste collection routes based on the real-time status of containers, taking into account both fill levels and air quality. Unlike static methods, this adaptive approach avoids unnecessary visits to low-priority containers, generating optimal paths that minimize total distance, travel time, and fuel consumption. The results show a significant reduction in these parameters, validating the effectiveness of deep reinforcement learning for intelligent urban solid waste management.

The elimination of redundant routes and prioritization of critical collection points translates into a reduced number of vehicles on the road. This directly contributes to a decrease in pollutant emissions (such as CO_2, NOx, and particulate matter) from combustion-engine collection vehicles. Additionally, fuel use is optimized, improving not only the operational efficiency of the service but also the environmental sustainability of public sanitation operations aligning with the principles of smart and green cities.

The proposed system is based on a modular distributed architecture, combining HC-SR04 ultrasonic sensors and MQ-135 air quality sensors connected to Heltec WiFi LoRa 32 V3 boards, communicating via LoRa to a central gateway. This setup, supported by a cloud database (Firebase) and a lightweight backend (Flask API), proved effective for near real-time data acquisition, transmission, and analysis. Its low energy consumption, long-range communication, and flexibility make it suitable for deployment in dense urban or hard-to-reach areas, with the ability to scale to more containers or add sensors without compromising system performance.

The model was pre-trained using simulated data with time acceleration, which allowed validation of its behavior and performance prior to physical deployment. This simulation phase, based on virtual sensors and real urban scenarios, enabled tuning of hyperparameters, refinement of the route selection policy, and assessment of the model in controlled conditions. The trained model was later deployed via a Flask API integrated with an interactive web application, demonstrating the feasibility of an end-to-end solution from data acquisition to visualization and field execution. This lays the groundwork for agile implementation in municipalities or other urban management entities.

Limitations and future work. Results were obtained in a controlled simulation with real geolocations and normalized sensing proxies; thus, field trials at scale are needed to assess performance under network variability (coverage, latency, packet loss), sensor faults, and urban traffic dynamics. Future work will extend the architecture to larger deployments, incorporate additional environmental variables, and explore multi-agent reinforcement learning for cooperative route planning among multiple collection vehicles.

BIBLIOGRAPHIC REFERENCES

1. Mr. Shreyas Bharadhwaj SP, Mr. Adithya S, Mr. Vishwas AM, Ms. Sharanya R C. Dynamic Route Optimization for Urban Waste Collection. Int J Res Appl Sci Eng Technol. 2025 Sep;13(4):1608–13. doi: 10.22214/ijraset.2025.68614

2. Jerbi H, Gnana Vincy VGA, Ben Aoun S, Abbassi R, Kchaou M. Optimizing waste management in smart Cities: An IoT-Based approach using dynamic bald eagle search optimization algorithm (DBESO) and machine learning. Journal of Urban Management. 2025 Jun 13;(13). doi: 10.1016/j.jum.2025.05.015

3. UNEP. Global Waste Management Outlook 2024 | UNEP - UN Environment Programme. 2024. https://www.unep.org/resources/global-waste-management-outlook-2024

4. Olawade DB, Fapohunda O, Wada OZ, Usman SO, Ige AO, Ajisafe O, et al. Smart waste management: A paradigm shift enabled by artificial intelligence. Waste Management Bulletin. 2024 Jun 1;2(2):244–63. doi: 10.1016/j.wmb.2024.05.001

5. Baldo D, Mecocci A, Parrino S, Peruzzi G, Pozzebon A. A Multi-Layer LoRaWAN Infrastructure for Smart Waste Management. Sensors 2021, Vol 21, Page 2600. 2021 Apr 7;21(8):2600. doi: 10.3390/S21082600

6. Stephan T, Hari Krishna SM, Lin CC, Sumesh U, Agarwal S, Kim H. ProWaste for proactive urban waste management using IoT and machine learning. Sci Rep. 2025 Jul 30;15(1):27790. doi: 10.1038/S41598-025-08452-X

7. Khallaf N, Abd-El Rouf O, Algarni AD, Hadhoud M, Kafafy A. Enhanced vehicle routing for medical waste management via hybrid deep reinforcement learning and optimization algorithms. Front Artif Intell. 2025 Feb 12;8:1496653. doi: 10.3389/FRAI.2025.1496653

8. Nguyen DVA, Gunawan A, Misir M, Hui LK, Vansteenwegen P. Deep reinforcement learning for solving the stochastic e-waste collection problem. Eur J Oper Res. 2025 Nov 16;327:309–25. doi: 10.1016/j.ejor.2025.04.033

9. Rabelo L, Rincón-Guio C, Laynes V, Gutierrez-Franco E, Bhat V, Zamora-Aguas J, et al. Effective last-mile delivery using reinforcement learning and social media-based traffic prediction in underdeveloped megacities. Discover Cities 2025 2:1. 2025 Aug 17;2(1):1–26. doi: 10.1007/S44327-025-00112-Z

10. Ramson SRJ, Vishnu S, Kirubaraj AA, Anagnostopoulos T, Abu-Mahfouz AM. A LoRaWAN IoT-Enabled Trash Bin Level Monitoring System. IEEE Trans Industr Inform. 2022 Feb 1;18(2):786–95. doi: 10.1109/TII.2021.3078556

11. Ayala A. NUEVOS RECOLECTORES DE RESIDUOS SÓLIDOS PARA LA CIUDAD. 2022. https://www.ibarra.gob.ec/site/2022/07/nuevos-recolectores-de-residuos-solidos-para-la-ciudad/

12. Cruz N, Cota N, Tremoceiro J, Davoli L, Martalò M, Ferrari G. LoRaWAN and Urban Waste Management—A Trial. Sensors. 2021 Mar 18;21(6):2142. doi: 10.3390/s21062142

13. Kavitha T, Chaganti KR, Elicherla SLR, Kumar MR, Chaithanya D, Manikanta K. Deep Reinforcement Learning for Energy Efficiency Optimization using Autonomous Waste Management in Smart Cities. In: 2025 5th

International Conference on Trends in Material Science and Inventive Materials (ICTMIM). Kanyakumari, India: IEEE; 2025. p. 272-8.

14. Yadav DK, M TAV, Aadil M. Open Lora Design and Implementation For IoT. In: 2023 International Conference on Power Energy, Environment & Intelligent Control (PEEIC). Greater Noida, India: IEEE; 2023. p. 822-6.

15. Croce D, Garlisi D, Giuliano F, Valvo A Lo, Mangione S, Tinnirello I. Performance of LoRa for Bike-Sharing Systems. In: 2019 AEIT International Conference of Electrical and Electronic Technologies for Automotive (AEIT AUTOMOTIVE). Turin, Italy: IEEE; 2019. p. 1-6.

16. Bhavanam BPR, Ragam P. Exploring LoRa Signal Propagation in Indoor and Outdoor Environments: A Comparative Study. In: 2024 IEEE International Conference on Smart Power Control and Renewable Energy (ICSPCRE). Rourkela, India: IEEE; 2024. p. 1-6.

17. Soto-Ortiz SI, Tolentino-Cruz MG, Porras-Rosas U. IoT monitoring and control prototype with Zabbix and LoRa communication, for Tilapia Biofloc systems. In: 2024 IEEE International Conference on Engineering Veracruz (ICEV). Boca del Rio, Veracruz, México: IEEE; 2024.

18. Wulandari A, Ramadhani EH, Fathurahman M, Hikmaturokhman A. Implementation Analysis of the PRESENT Algorithm on ATmega328P for Securing RFM95W LoRa Communications. In: 2024 IEEE International Conference on Communication, Networks and Satellite (COMNETSAT). Mataram, Indonesia: IEEE; 2024. p. 615-21.

19. Liu Y, Cai J, Qiu M. Research on low-power and low-cost smart animal husbandry positioning system based on LoRa transmission. In: 2025 5th International Symposium on Computer Technology and Information Science (ISCTIS). Xi'an, China: IEEE; 2025. p. 571-5.

20. Sujatha V, Prasanna KL, Niharika K, Charishma V, Sai KB. Network Intrusion Detection using Deep Reinforcement Learning. In: 2023 7th International Conference on Computing Methodologies and Communication (ICCMC). Erode, India: IEEE; 2023. p. 1146-50.

21. Jayakody D. Deep Q-Networks (DQN) - A Quick Introduction (with Code). 2022. https://dilithjay.com/blog/dqn

22. Caiza G, Soto-Rodríguez A, Ayala P, Garcia CA, García M V. Applying Deep Q-Networks to Local Route Optimization. In: 2024 IEEE 22nd International Conference on Industrial Informatics (INDIN). Beijing, China: IEEE; 2024. p. 1-8.

23. Heltec Automation. WiFi LoRa 32(V3). 2024. https://heltec.org/project/wifi-lora-32-v3/

FINANCING
The author(s) received no financial support for the research.

CONFLICT OF INTEREST
The authors declare no conflicts of interest. The funders had no role in the study design, in the manuscript writing, or in the decision to publish the results.

AUTHORSHIP CONTRIBUTION
Conceptualization: Fabián Cuzme-Rodríguez and Gabriela Caragulla-Briceño.
Data curation: Francisco Ger-Ortega, Gabriela Caragulla-Briceño and Luis Suárez-Zambrano.
Formal analysis: Luis Suárez-Zambrano and Henry Farinango-Endara.
Research: Gabriela Caragulla-Briceño, Francisco Ger-Ortega and Alejandra Pinto-Erazo.
Methodology: Fabián Cuzme-Rodríguez and Luis Suárez-Zambrano.
Project management: Alejandra Pinto-Erazo and Henry Farinango-Endara.
Resources: Gabriela Caragulla-Briceño, Francisco Ger-Ortega and Fabián Cuzme-Rodríguez.
Software: Gabriela Caragulla-Briceño and Francisco Ger-Ortega.
Supervision: Fabián Cuzme-Rodríguez and Luis Suárez-Zambrano.
Validation: Alejandra Pinto-Erazo and Henry Farinango-Endara.
Display: Gabriela Caragulla-Briceño and Francisco Ger-Ortega.
Drafting - original draft: Gabriela Caragulla-Briceño, Francisco Ger-Ortega and Fabián Cuzme-Rodríguez.
Writing - proofreading and editing: Fabián Cuzme-Rodríguez and Gabriela Caragulla-Briceño.