Simulated LoRa Sensor Network as support for route planning in solid waste collection*

Montañez Gomez, Miguel Angel
Facultad de Ingeniería – Depto. de Ingeniería de Sistemas
e Industrial. Universidad Nacional de Colombia, Bogotá
mmontanezg@unal.edu.co

Niño Vasquez, Luis Fernando
Facultad de Ingeniería – Depto. de Ingeniería de Sistemas
e Industrial. Universidad Nacional de Colombia, Bogotá
lfninov@unal.edu.co

ABSTRACT

Mass production and population growth have produced an increase in the generation of municipal solid waste (MSW) in urban settlements. In consequence, efficient treatment of waste has become a challenge for cleaning entities, possibly because they continue performing collecting operations using fixed periodic routes that exhaustively go across neighbourhood streets in search of every dumpster. Furthermore, in these operations recyclable material is not separated from disposable one, at least in Bogota, thus causing a negative impact on the environment. This work aims to prototype a sensing system that generates routes based on the actual fulfilment level reported by dumpsters. For this purpose, dumpsters were equipped with a level measurement device that uses a proximity ultrasonic sensor. Information was transmitted using a LPWAN (LoRa), and collected data were used to determine which of the dumpsters needs to be collected in the route planned for a specific date and which ones can be deferred until the next collection, according to the levels reported. Since an operation requires many collecting trucks, a K-means method is used to group dumpsters that are geographically close. A single district of Bogota was selected for demonstration purposes. The collecting sequence was calculated using an open Web service, whose results are shown on an Android mobile application. The mobile app uses the Google Maps routing service. This demonstrated technical viability for implementing this sort of systems in real operation environments leading to reduce container saturation and overflow

KEYWORDS

IoT, Waste Collection, wireless sensor networks, LoRaWAN, sensing system, CVRP, geographic information, mobile application**.**

1.   Introduction

Globalization and industrialization of factories have significantly reduced the cost of products, thus giving us access to all kinds of goods ranging from clothes, to electronic devices and house equipment, among others. This low cost offer has changed society consumption habits, which increased the amount of products that are sold, used, replaced and discarded in small periods. Additionally, non-biodegradable packaging used to protect products and the fast growth of population induced massive amounts of MSW production. Indeed, North America waste production has duplicated between 2006 and 2010. According to a report of the World Bank (Kaza et al, 2020), 2.01 billion tons of MSW are generated worldwide each year, and at least 1/3 of that waste is not properly managed.

Non treated solid waste is endangering ecosystems, producing greenhouse gases, and contaminating ecosystems. These effects have a negative impact not only on human's health and life style, but also on animal habitats and ocean ecosystems. Then, an efficient treatment of solid waste management has become a critical aspect.

An efficient treatment of MSW encompasses at least 3 aspects: the reduction of waste generation, a structured recycling process and a proper disposition of non-reusable waste. Waste classification and transportation is an important step in the recycling process, and commonly city governments are responsible for collection logistics.

Campbell & Wilson (2014) explain how collecting MSW has been treated with a deterministic approach using periodic routes for more than 40 years. This problem is known as periodic vehicle routing (PVRP). Under this approach, collector trucks travel along fixed routes one or more days per week, repeating same routes every week. This approach was useful until a few years ago, but now it has shown to be inefficient, considering the increase in MWS production in big cities, such as Bogota. In some districts, solid waste overflows the container capacity and ends dispersed on the ground. This produces aesthetic problems of city cleanness that as a consequence of rotten food and their bad smells might attract

insects or rats that transmit infections, thus turning the situation into a public health problem. Ospina-Pinto et al. (2017)

In this research, we address a way to structure the logistic process of MSW collection according to the necessities of a single district of Bogotá, trying to achieve an efficient collection operation that reduces container saturation and overflow.

The main contributions of this work are:
i.   Assembling three different configurations for measure devices, using a long distance transmission technology.
ii. Testing and comparing transmission ranges of assembled devices.
iii. A simulation of waste generation in the streets of a district in Bogota, using simulated information to generate routes dynamically considering real-time traffic.
iv. A mobile application that shows container fulfillment levels, containers locations and shows generated routes on a map.

The remainder of this paper is organized as follows. Section 2 presents a brief review of related work. Section 3 outlines an overview of the system architecture, then, Section 4 describes the system implementation, including hardware integration, data collection and transmission, transmission range distance test, data simulation, route generation and mobile application demonstration. In section 5, we discuss results and system performance. Finally, conclusions and future work are devised in section 6.

2.   Related Work

Trying to stay sustainable and considering citizen well-being, governments of the cities strive to find an efficient way to collect and manage MSW. To deal with this concern, many studies have been proposed. For instance, (Babaee Tirkolaee et al., 2019) proposed an efficient simulated annealing (SA) applied to MSW collection in Sanandaj, Iran. Heuristics-algorithms using python scripts with ArcGIS were proposed in (Amal et al., 2018) and tested in Sax, Tunisia. In South America, an example takes place in Lima, Peru, where Gilardino, Rojas et al. (2017) suggested a mathematical optimization model to reduce number of garbage compactor trucks trying to reduce greenhouse emissions.

Other solving approaches include a two-phase memetic algorithm that uses clustering and sequencing for solving VRP with time-windows (Bustos Coral et al., 2018), swarm optimization (Abdulkader et al., 2015), genetic algorithms (Fujdiak et al., 2016), or an hybrid of both (Kuo & Zulvia, 2017). These studies show important results, the first is reducing distances and collection times, and the second is a fast solution convergence, which is important considering VRP is an NP-hard problem.

Routing of vehicles using Geographic information systems GIS and network analytics has also been studied. (Chaudhary et al., 2019)

and (Jwad & Hasson, 2018) used the ArcGIS network analytics tools to define collection zones and streets paths. In their study, fuel usage, distance, time and number of vehicles were quantified as variables in an efficiency equation to evaluate the solution performance.

These works offer solutions in terms of static routes, which underestimates the fact that waste generation is a non-deterministic phenomena. Then, using static routes might reduce distance, time or even the operation cost, but they are not estimating the real necessities of the city. One of the difficulties to do that estimation is the lack of updated information in relation to solid waste generation. This problem was identified by (Medvedev et al., 2015), who highlighted the importance of providing elements of collection systems with certain degree of perception, using IoT components like sensors, RFID or cameras to get real time information. The use of IoT can lead to an optimized collection based on the information reported by components. This vision completely changes the way we understood MSW collection and open an IoT perspective that has being widely accepted and researched by the scientist community.

In this boom of IoT solutions for waste generation sensing, low cost ultrasonic sensors HC-SR04 are commonly used to measure the filling level. The main difference is data transmission technology, e.g. GPRS is used by (Yusof et al., 2018) to communicate dumpster levels as notifications direct to mobile terminals. With this approach little data processing is done to allow significant decisions.

A more complex and complete sensing system was proposed in (Pardini et al., 2020), in which the measurement device integrates a load sensor HX711, temperature and humidity sensor DHT11, and GPS module NEO6M. Apart from the ultrasonic sensor, the circuit was assembled on an Arduino board. Data is transmitted using a GPSR module SIM900 which operates in 2G cellular technology. The processed information is delivered to citizens through a mobile application, thus calling for community interaction to receive feedback that can be applied to enhance waste collection planning.

However, GPRS results to be quite expensive in terms of energy usage, so in (Karthikeyan et al., 2018) they implemented a wireless sensor network WSN using Zigbee as transmission technology. Zigbee uses a particular topology in which measurement devices communicate with coordinator nodes. These coordinators serialize information and transfer it to a gateway built on the top of a Raspberry PI. An MQTT broker was used to manage the data between the network and the applications. This work is very useful as it reports transmission ranges and concludes that Zigbee shows 97% reliability in distances shorter than 80 meters, but exceeding this distance the signal was considerably attenuated and packages got lost.

A similar work was published in (Chaudhari & Bhole, 2018). In this work, the selected transmission technology was Wi-Fi, and the coverage distances were very close to the ones reported with the use of Zigbee, i.e. between 100 and 120 meters. This work uses a mobile application to show the location of containers that exceed the defined threshold on a map, so the user interface is much explicit and easier to understand. A very similar solution was published in (Kang et al., 2020), but it was oriented to deal with electronic waste exclusively.

The distance limitations can be tackled using LPWAN (Low Power Wide Area Networks), which is a set of technologies capable of covering wide areas with few devices. A relevant research of LPWAN applied to waste management systems is presented by (Lozano et al., 2018). In this case, information is transmitted using LoRa modulation, since the authors declare that the LoRa signal can deliver data in a range from 5 to 15 Km, consuming around 0.5 µA. They were able to deploy a WSN for the rural area of Salamanca and evidence an important improvement in reducing collection cost compared with the static collection approach.

An important aspect to consider in a WSN is energy efficiency. As we expect to give the least possible maintenance to end nodes, constantly replacing batteries may become a problem for network maintainability. A comparison among three of the most popular transmission technologies was made in the context of agricultural monitoring: Zigbee, WiFi 2.4 GHz, and LoRaWAN. The results show that LoRaWAN has a better yield when power consumption and transmission distances were the priority. (Sadowski & Spachos, 2020)

Besides the right technology, it is critical to consider how end devices manage energy to have a longtime working network. The node architecture presented in (Cerchecci et al., 2018) extends battery life on end device. They attached a low consumption timer and a trigger to the node. The idea behind this architecture is that after the main circuit obtains the measure, sends the information and gets a confirmation from the server, it sends a signal to start the timer and switches down the measurement circuit. The timer will count until the next measurement cycle is required, consuming 4µV. When the cycle is completed, it triggers again the measurement circuit avoiding it to be active the whole time, misspending energy.

LPWAN has been applied to several fields, showing high performance and reliability to transfer data and fault tolerance. Some interesting applications include: livestock remote surveillance based on a collar to monitor vital sings in cows (Li et sl., 2018); metering the average consumption of households in the electrical grid in cities (de castro tomé, nardelli, & alves, 2019), and remote sensing humidity of soils and control field irrigation in agriculture (Zhao et al., 2018). LPWAN has also been used to reduce the cost of public street illumination by adjusting brightness, detecting abnormal consumption and determining when a streetlight needs maintenance (Bingol et al., 2019). An ad hoc network was implemented in a bicycle shared system in Lyon-France, to share data. Bicycles are equipped with a transmission module and bike stations are bridged to internet to send data collected by bicycles (Zguira et al., 2018)

Once the IoT server receives data, it processes the information in such a way that efficiently computing routes becomes a challenge of high complexity because finding routes in multiple paths is a combinatorial problem. In the case of waste collection, trucks have a limited load capacity, they should go along a set of locations picking up the waste material until their load capacity is completed. The material has a certain weight or volume, but this should never exceed the capacity the truck can carry. This issue is known as the capacitated vehicle routing problem CVRP (Faulin et al., 2011). to solve this, (Medvedev et al., 2015) suggested a simple top-K query method that executes a query on a relational data base and selects the K containers reporting the highest filling levels, then, container coordinates Lat/Long are sent to a Google Service that delivers the best routes, taking into account streets configuration.

If locations are updated constantly, routing can be carried out with a combinatorial General Variable Neighborhood search as presented in (Papalitsas et al., 2018). With this method, containers are organized by insertion according to their characteristics. They are placed near to containers with similar latitude and longitude. Similarity is measured using Euclidean distance, so containers are organized in ascendant order in order to cluster containers spatially close. Once all the selected containers are inserted in the list, they are separated by preserving order and considering truckload capacity. The algorithm was coded in Fortran and validated with the benchmark offered by Heidelberg University in TSPLIB (Heidelberg University, 1997)

This literature review led us to propose an IoT solution that can be an interesting alternative to periodic routes for MSW in urban areas. It also helped us to define a transmission technology that adapts better to Bogota conditions, considering the area to be covered, signal integrity and the elements that need to be well used, such as the battery and data transmission coverage. We designed a system architecture that is explained next.

M. Montañez & L. Niño.
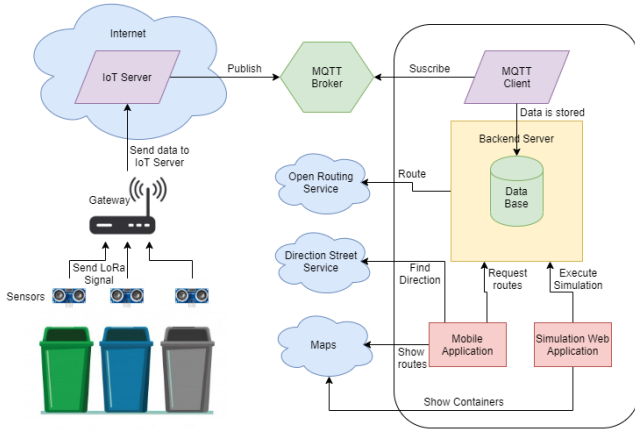
3. System Overview and Architecture

Figure 1: General configuration and communication of system components

The proposed system is arranged as shown in Figure 1. The filling level is sensed periodically. Containers are indexed in a database with a unique identifier and geographical location (Lat, Lon) in such a way that, as data is received, it is possible to know the approximate location of the container whose filling level is being reported. These data are transmitted using the LoRaWAN protocol. LoRa Signal is decoded in the gateway and broadcasted to the internet, where it gets routed to an IoT Server where data is managed, a server appliance publishes these data to a topic in a MQTT broker.

Once the information is registered in the broker, it can be queried via an MQTT client running in a Backend server that subscribes to the topic and retrieves data stored in a database so that applications can access it when required. Given the limited amount of measurement devices for the selected area a simulation was required to execute routing with significant data. Simulated data is used to plan daily operation, which means streets are given a priority based on the level reported by the containers located in the street priority rules explained in section 6.

The estimated volume of MSW to be collected in the zone, allows to decide the number of trucks required to pick that volume. A clustering method is executed in order to group streets that are geographically close, so computing time in routing operations could be reduced. Routing was managed using an Open Web Service.

Calculated routes are displayed using a mobile app, which demands generated routes to the backend server and then it shows the corresponding streets. The trucks should go through the locations of dumpsters that need to be served. A map provider was

integrated so that the information of routes can be easily interpreted. It also uses a direction service that indicates the way to each truck, considering real-time traffic information to prevent the trucks to turn around or to be stuck in traffic jams.

4. Methods and Results

4.1. Hardware Integration

Measuring the level is possible using a weight sensor, however, there are some debates about their use, since depending on the material density the container might report high weight without being full, or vice versa. For that reason, we decided to use a distance sensor HC-SR04. The operation principle behind this sensor is to emit an ultrasonic signal that travels through the air until it collides with the closest obstacle, then, the signal is reflected and the sensors receive the signal back. Distance is estimated by measuring the delay between the signal emission and when it is acquired back, as explained in formula 1.

$$Distance = \frac{(Speed \ X \ Delay)}{2} \quad (1)$$

The sensor returns back the time. As we know that sounds propagate at 334 m/s in the air, then, 0,0334 m/μs is divided into 2 because the signal spends time to go and return. Afterwards, we can calculate the distance.

To make hardware assembling easier, development boards were used integrating analogical and digital pins. For conceptual reasons three configurations for measuring units were explored and analyzed. All these configurations are described in the block diagram in Figure 2.
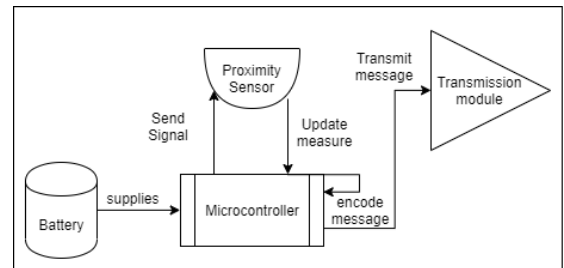

Figure 2: Block diagram for Measuring unit components

The first configuration was made on the top of an Arduino Mega Board with a LoRa Shield created by Dragino. Both devices were connected according to the pins mapping and digital pins 30 and 31 were used to connect proximity sensors as shown in Figure 3-A. Table 1 lists pin interfacing between HC-SR04 and Arduino MEGA.
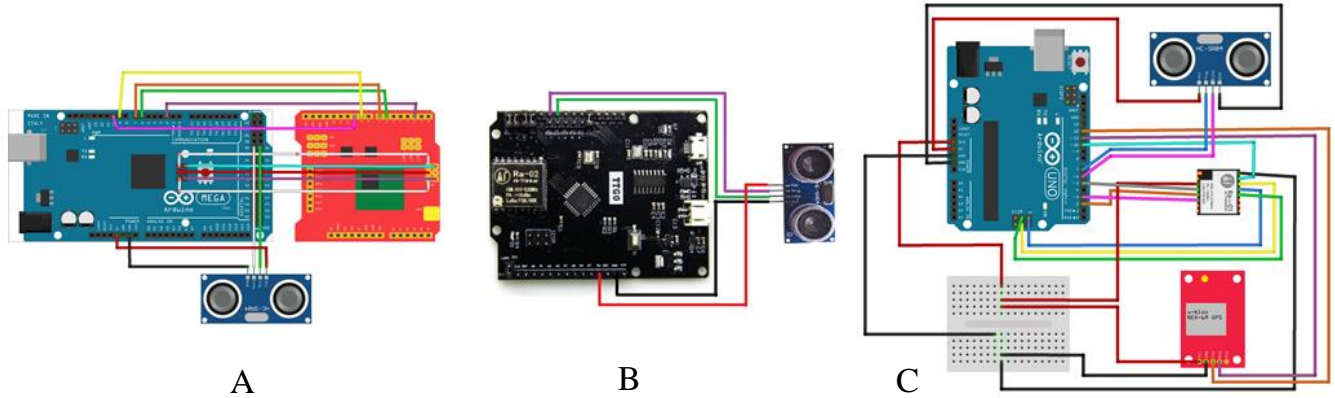
Figure 3: Diagrams of measurement devices connection: (A) interfacing for Arduino Mega and Dragino, (B) interfacing for TTGO and HC-SR04, (**C**) interfacing for Arduino, SX1276, Neo 6M and HC-SR04

Arduino	HC-SR04
5 V	5 V
GND	GND
D30	Echo
D31	Trigger

Table 1: Pin mapping of Arduino Mega to HC-SR04

The next circuit was set using a board that integrates LoRa transmission module. This simplifies the connection of the component, since the only component that needs to be added to the board was the proximity sensor, as shown in Figure 3-B. Table 2 states pin interfacing between HC-SR04 and TTGO.

TTGO	HC-SR04
5 V	5 V
GND	GND
D6	Echo
D7	Trigger

Table 2: TTGO to HC-SR04 pin mapping

The third circuit has a small difference compared with the previous ones, as this one integrates a GPS module (U-blox neo 6-m), so that geographical location is obtained precisely. This can help to be more accurate when planning the route. However, this circuit has significant impact on energy efficiency. Besides, as the container locations remain almost constant, updating the location each time a measure is taken is not recommended. The circuit was assembled as shown in Figure 3-C.

Arduino	SX1276
3,3 V	3,3 V
GND	GND
DI0	D2
DI1	D3
MISO	MISO
MOSI	MOSI
SCK	SCK
D10	NSS
D5	Reset

Table 3: Pin mapping of Arduino UNO to SX1276

Arduino	Neo-6M
3,3 V	3,3 V
GND	GND
D11	RX
D12	TX

Table 4: Pin mapping of Arduino UNO to Neo-6M

Connections between Arduino UNO and HC-SR04 are the same that stands in table 2. The controllers were programmed using a LoRa Library for Arduino written by Kooijman (2020). The library implements a listener method, which is responsible of establishing a communication session with the IoT server after the joining request is accepted. Then, the distance measured by the sensor can be encoded and transmitted.

4.2. Data collection and transmission

Now that data was captured and transmitted by measure units, we expect to obtain these data remotely, so it is necessary to configure the gateway, the server and the broker.

For the gateway component, we found a commercial solution that can gather data transmitted with Lora modulation in a particular frequency. The selected reference was LG02 from Dragino. This gateway offers a graphical user interface (GUI) to set up all transmission parameters, such as spreading factor – SF, band width – BW, frequency, coding rate and IoT service provider. The parameters selected for our reception and transmission channels are shown in Tables 5 and 6, respectively. This gateway has two communication channels that allow it to operate in full duplex[1].

Channel 1	Reception
Frequency	902320000
Spreading factor	7
Coding rate	4/5
Signal bandwidth	125 khz
Preamble length	8
Lora sync worD	52

Table 5: Parameters for reception channel

Channel 2	Transmission
Frequency	923300000
Spreading factor	7
Coding rate	4/5
Signal bandwidth	125 khz
Preamble length	8
Lora sync word	60

Table 6: Parameters for transmission channel

After completing gateway configuration, it was started in listening mode and connected to the network using the Ethernet interface.

For the IoT server, we decided to use the things Network[2] TTN because it supports LoRaWAN protocol operations such as activation by personalization (ABP) and Over the Air Activation (OTAA) as authentication methods. TTN simplifies the implementation of decoding and validation rules for the information being reported by the nodes. Consequently, information can be filtered before doing any post-processing operations.

The very first step to use the server is to register the gateway via TTN. The administration console shows some information required such as identification code, frequency plan and the type of antenna that the gateway has attached. After the gateway is configured, it is shown in the console (see Figure 4). Notice that when the server established connection with the gateway, the status is set to "connected".
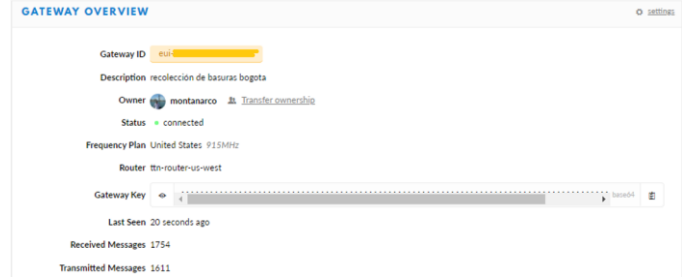

Figure 4: Gateway registry on the IoT server.

We also configured an application that is intended for two purposes. The first is to associate all the measure nodes that will be part of the network. The second is to define general rules and procedures that will be applied to the information reported by the nodes associated to the application, so that information can be managed in a generalized way, e.g., decode functions and integrations.

In our case, the measures taken in the remote devices are encoded in an array of 8 bytes length, bytes 0 to 2 were used to encode latitude, bytes 3 to 5 were used to encode longitude and bytes 6 and 7 were reserved to the filling level, so we used a decode function that transforms this 64 base byte array into a JSON format, which is more understandable by users.

Then, the integrations phase defines how to process data after it is received and validated in the server. We decided to use a MQTT broker because the server implements a Mosquito[3] instance and creates a topic with the name of the application configured. So that whenever new data coming from measure units reach the server and are successfully evaluated according to the validation and decoding functions, the readable payload is published in the topic created for the application in the broker. As soon as the IoT server was configured, it was started along with the gateway, so both of them remain up and running.

The following step was to extract data from the broker, hence an MQTT client was implemented and ran in the backend server. We used a Java implementation offered by TTN. The repository is open and it is available in a public repository (Cambier, 2016). The source code has a listener that continuously demands for information to the broker. When new information is obtained, we use a function that processes these data, by searching the id code of the dumpster that sends the information, then, that data is stored in a Postgres data base.

---

[1] Full duplex is a communication mode that allows to emit and receive messages simultaneously.

[2] An IoT server designed for LPWAN available at https://www.thethingsnetwork.org/
[3] An open source lightweight MQTT broker supported by Eclipse foundation

4.3. Transmission results

After some configurations, we ran a transmission test. Figure 5-A show the log running on the gateway, with encoded messages being reported by nodes and captured and retransmitted by the gateway.

A

B

C

Figure 5: Messages flow through network components: (A) messages received in the gateway show in the Log read, (B) messages received and validated by the Application in the IoT server, (C) messages retrieved by MQTT client running in the Backend Server

Figure 5-B is a screenshot of the data application interface showing messages that where correctly decoded and validated according to the rules defined for the application. Notice that messages from the three configured end nodes are received in the server.

Figure 5-C is the log of the MQTT client running on the back end side and taking the new messages form the application topic in the broker. At this point, information has already been decoded, so that the payload can be read. Two elements are recognized, distance and location Lat/Long, then, the application matches the reported information to their correspondent dumpster based on code used to index it on the database.

Figure 6: Data stored in the measures table after retrieval by the MQTT client.

Figure 6 is the result of a query on the data schema that shows the last measured reported by the nodes connected to the network. The data remains available so when any application needs it, it can be directly queried to the database. A simple web application was implemented to visualize the data collected by measure units (see Figure 7).

Figure 7: Web application displaying data reported by measurement units.

The first result to highlight is that information flows along the LoRa network delivering data from measurement unit to the backend database, as expected.

We carried on the research executing a couple of tests to verify the real distance reached by the selected transmission modules. The gateway was started and placed in a fixed location, then, devices were configured to send a message every minute. These data were stored in the database and represented in a map, and the longest reached distance was measured.
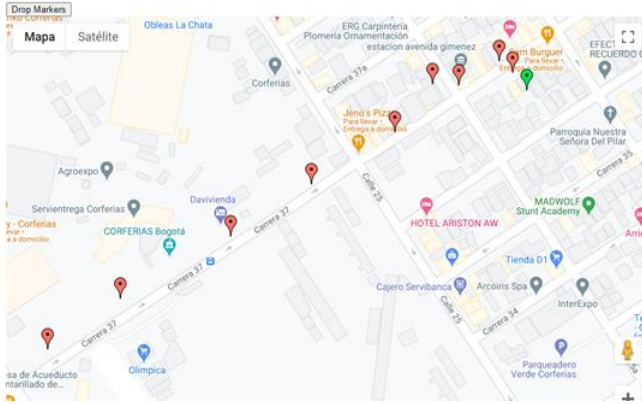
Figure 8: Transmission test using Arduino circuit: (A) locations reported by Arduino in the transmission test, (B) maximum range reached with Arduino measurement unit.

The first test results are shown in Figure 8-A, where the green marker represents the gateway location, while red markers indicate locations retrieved by the GPS in the measure unit, reported in the server and stored in the database. We selected the green marker and the furthest red market locations, after using Google Maps to approximate distance. The longest distance was about 488 meters (Figure 8-B).

This test evidenced that signal did not reached the gateway when there was an obstacle, e.g., a wall that interrupts the direct line of sight. Thus, we concluded that an effective transmission requires a direct line of sight between both devices. Consequently, for the second test, we decided to find a high-altitude location to place the gateway, then, we started walking away from the gateway in a low-density building area so that the direct line of sight was easier to achieve.
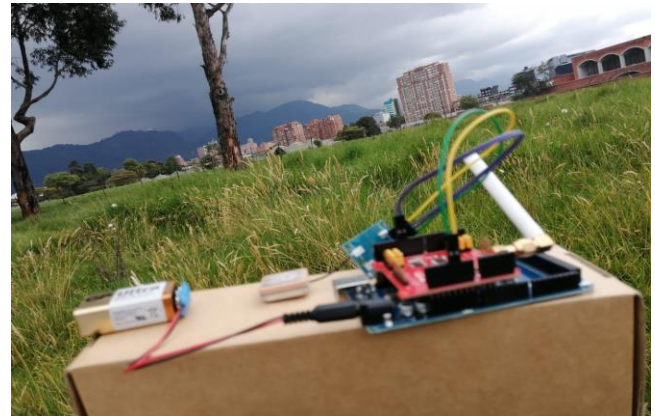
Figure 9: Dragino Unit executing data transmission test.

Figure 9 shows the Dragino device used to execute the test. The gateway was located in the 13th floor of the building shown in the upper right side of the picture. This test showed longer distances than the previous one, specifically 1.39 km (Figure 10-B).

A point to consider is that range coverage can be extended in dense urban areas by augmenting the SF, according to (Sagir et al., 2019). This happens because high SF extends airtime signal, thus giving more chances to the receptor to receive the message. This configuration reduces the data rate.
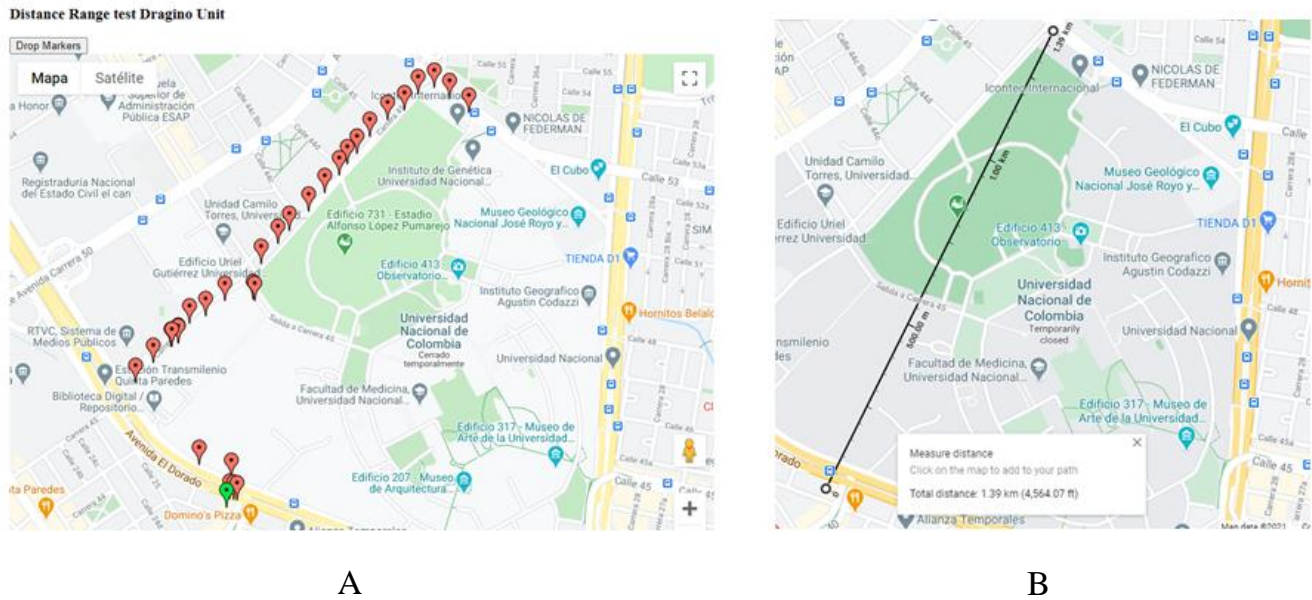
A

B

Figure 10: transmission test using Dragino circuit: (A) locations reported by Dragino in transmission test, (B) maximum range reached with Dragino measurement unit.

4.4. Data Simulation

As we only provide three real measurement units, it is impossible to generate other significant plans and routes for collecting MSW than visiting the locations of our three measuring units. For this reason, it was required to simulate a greater number of units reporting filling levels in order to evaluate route calculation under more complex conditions.

To simulate container locations, we defined a set of rules:

1. a container should be located on a street where a truck can reach it.
2. containers are located always in pairs: one for general purpose waste and another for recyclable material.
3. Each pair of containers should be separated from the next one by at least one street length distance, i.e., 100 meters.

The third rule was conceived to find an even distribution of containers thus avoiding many containers to be crowded in the same area. To implement the first rule, it was necessary to have geographical information of street configuration in the study region, therefore, we downloaded geographical data from "Unidad Administrativa Especial de Catastro Distrital Bogota" - IDECA. The first set of geographical data contains the street axis lines that shape the road network of Bogota, and the second one specifies the blocks and neighborhood belonging to the different Bogota districts.

The downloaded information was treated and organized. The simulation was divided into two phases. The first one was to put representations of empty containers in a street where a truck could reach them and find an even distribution along the study area. For this purpose, we used the street locations and found a middle point in the street coordinate. The generated containers were represented in a map, as shown in Figure 11-A.

The second phase was to update the level of containers using random numbers between 0 and 100 that represented the percentage of filling. These numbers have the same mean and standard deviation found in a dataset of waste collecting routes from Austin Texas, downloaded from Kaggle[4]. The color of each container was updated according to the convention shown in Table 7. The result is shown in Figure 11-B.

level	Priority	Color
Less than 10%	empty	black
Between 11% and 30%	low	green
Between 31% and 55%	medium	yellow
Between 56% and 80%	high	purple
Greater than 81%	full	red

Table 7: Colour code for representation of container priority

---

[4] this dataset is offered by Austin government it includes information of Austin waste and diversion 2008-2016 (Boysen, 2017)

This filling percentages were defined based on the test executed by (Meesala et al, 2018), which was validated using a measurement device that integrates an ultrasonic sensor and considers the directive angle of the sensor signal.
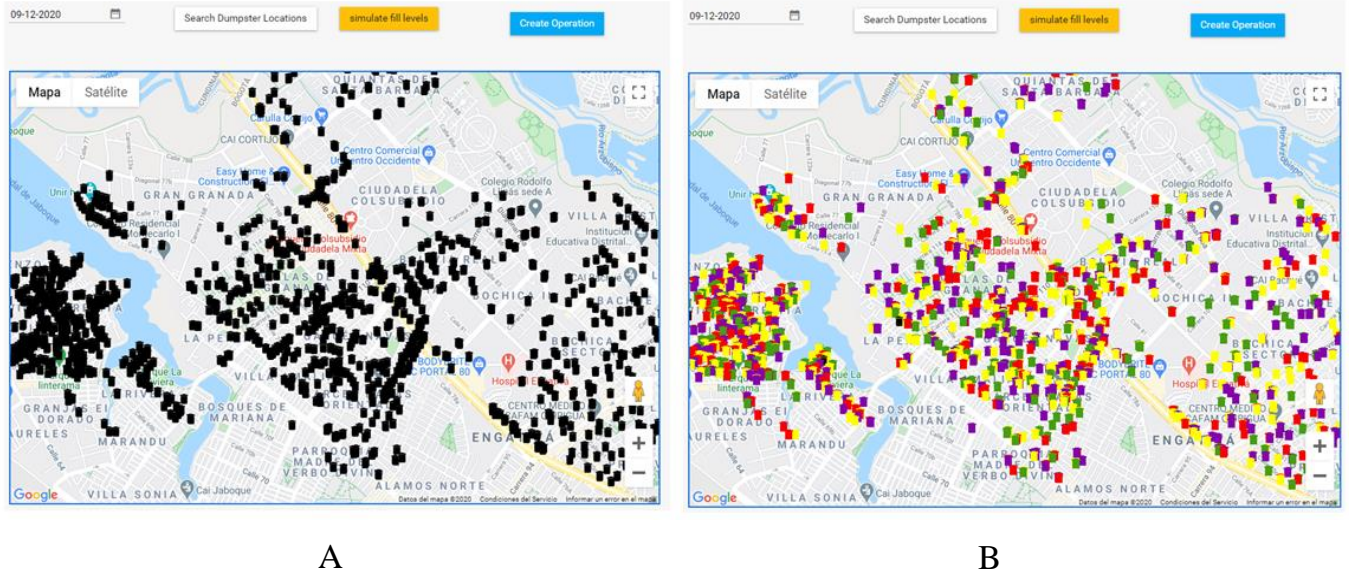

Figure 11: MSW simulation in Engativa District: (A) phase one container location simulation, (B) phase two container filling level simulation

Street Priority	Dumpster level reported	Description
1	full > 0	There is at least a full container, or the container has not been collected in more than 3 days.
2	high > 0	There is at least a high-level container, or the container has not been collected in more than 2 days.
3	Medium > 0	There are only containers with medium and low levels.
4	low > 0	There are only containers with low level.

Table 8: Rules to define street priority

We considered some facts looking for an efficient operation. For instance, if the truck must go across a street to collect the content of a high level container, it must also collect other containers in the same street. The reason is that, when the truck collects only the container that reports the high level, on the next days it should return to the same street to collect other containers, thus increasing the travel distance.

Also, it is required to reduce streets distance between containers, because sometimes containers may seem spatially close, but the streets connecting them are not. Therefore, in many cases containers that are spatially farther, but their travel distance is shorter must be collected first.

Taking these observations into account, it results easier to prioritize the streets where containers are located on, rather than containers themselves. Then, we established the street priority based on the rules defined in Table 8.

With clear information of the streets prioritized and the containers that must be collected, we were almost ready to start planning the collection operation. However, route calculation can be computationally expensive and time consuming, therefore, for routing vehicles we will use a heuristic search based on tree spanning. Since we know the approximate volume of MSW that should be collected and the capacity of trucks in the fleet, we can estimate the number of trucks required for the operation. Even more, knowing the location of containers allows grouping the ones that are geographically close in such a way that those containers can be served by the same truck.

K-means Clustering was applied in MSW collection planning, as one of the important prepossessing tasks before routing by (Vu &

Kaddoum, 2017), using data of Philadelphia, Pennsylvania, USA. Using geographical coordinates, latitude and longitude, as grouping feature, they applied Elbow method as convergence strategy and a logistic regression to forecast the dumpsters that will probably be full to include them in the operation. Taking some elements of this research, we decided to use K-mean as clustering method, as fully explained in (Nazeer & Sebastian, 2009). The fundamentals steps behind this method are:

1. To define a feature reference and a number of groups.

2. Initial groups of centroids are defined randomly, then, elements are assigned equitably to the groups. One element is assigned to a group whose centroid is closer with respect to the value of the element itself.

3. The group centroids are recalculated by extracting the means of the elements assigned to that group, and the elements are relocated in new groups according to the new group centres.

4. The operation is iterative and it repeats until there is no reduction in global distances.

Global distances are calculated as the sum of the distances of each element to the centroid of the group where it was assigned. By reducing distances in each group, the global distance is also reduced.

For clustering, we implemented a function that sums the volume of MSW located in the streets that have priority 1 or 2. Using this volume, we estimate K as the number of trucks required for the operation. That number matches the number of groups. So, after grouping, each truck could be assigned exactly to one group, thus, we include a constraint to avoid that the accumulated volume of a group of streets exceeds the volume that the one truck can carry. Then, we extracted the centroid of those streets in terms of latitude and longitude, and we use these values as reference features. The result of streets clustering is shown in Figure 12.
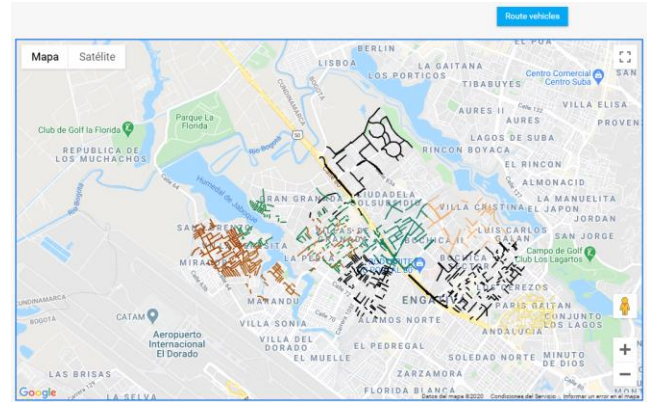

Figure 12: Prioritized streets clustering with K-means method using K=9.

4.5. Route Calculation and Delivery

Once streets were prioritized and grouped, the next step was to find a collection sequence that reduces travel distance in each group. Taking advantage of the available services, we decided to use one applied to route vehicles. Open Route Service (ORS) uses a routing strategy called tree spanning, which consists in finding the way of connecting nodes in different locations to a central node. An economic way to do so is by connecting all nodes in one direction by a single line, in such a way that, all nodes connected in the same direction can use the same path to arrive to the central node. The routing cost for 2 vertices is given as the sum of the cost of the edges in the unique tree path between them (Gouveia & Martins, 1999). The strategy was originally proposed for communication routing. Nonetheless, considering the streets distance or time as a factor of the route cost, it can be applied to vehicle routing. (Wu et al., 2000) The service performs some modifications to the initial solution trying to reduce distances by applying operators. Some of them are: relocation, interchange, inversion and alteration. The solution strategy was described in detail by (Zhou, et al. 2018). The service developers have published a video explaining the service solution strategy, functioning and use. This video can be accessed in (Coupey, J. 2018).

As ORS is offered through an API, getting a key to use it was compulsory. The service we used was optimization, which requires to specify the places to visit and the truck features in a structured way in JSON format, documented in an open repository[5].

K-means grouping represents a significant progress in two ways: first, the assignation allows us to know in advance which truck will serve a group of containers, and second the centroid of each group may work as a start location for each truck. Therefore, when we

---

[5] Project is open-source optimization engine provide solutions to various VPR in within a small computing time (Sharma, 2016)

execute a routing task with the service, we assign exactly one truck, then, we transform each container location that belongs to the group into a job using the format that the API expects. As a response, the service returns another JSON containing a list of trucks with an array of container locations with high priority that the truck must visit, in a specific order that reduces the travel distance, then, the solution is stored in the database to avoid reprocessing every time an application demands routes. Accordingly, the routes information is available as soon as external applications require it, thus saving time, but also considering the service constrains of use per day.

Up to this point, we have the routes described in terms of geographical coordinates, now we need an effective way to display such information to users in an easy to read and understand format. It is likely that the people who use this information will be truck drivers while searching dumpsters, hence it would be comfortable to visualize information in a portable device with internet connection. For this reason, we considered that a mobile application could fit these requirements.

A mobile application was coded in Flutter, a recent Google framework for mobile application development, which has an easy integration with Google services, such as Google Maps. We used a REST architecture for the application simply exposing services to manage the request made by client applications. This is a simple way to transfer information in JSON format: some of the services exposed were authentication and generated routes in a specific date.

The application demands user authentication; this user has a role with permissions to see all the generated routes. The list of routes is queried via "get verb" using the date as parameter. The response is a list of routes displayed in a component called "List view" (see Figure 13).
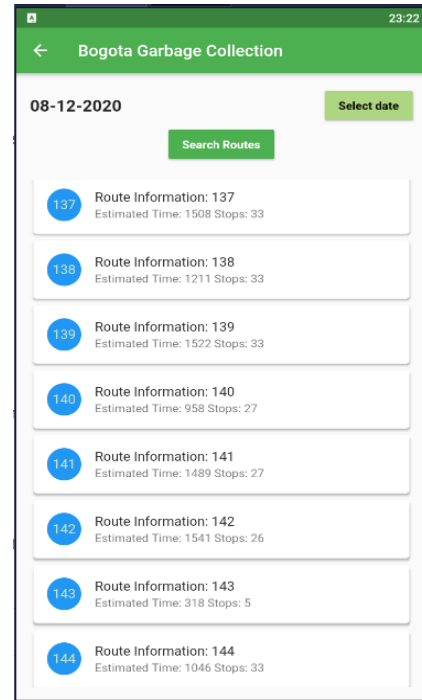

Figure 13: List of routes planned for 08 December 2020

By tapping one element of the list, the application will display the locations of containers that must be served in the current route (see Figure 14).
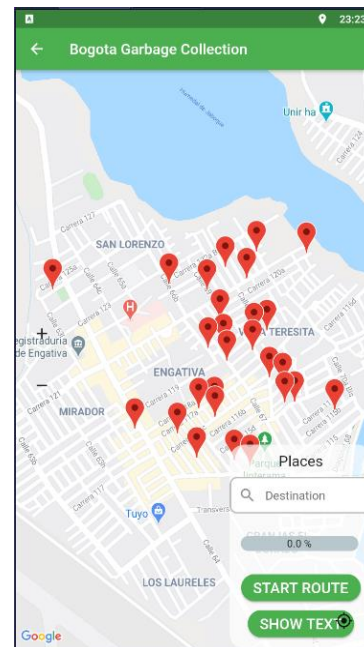

Figure 14: Initial view of containers when a route is selected

When the *start route* button is pressed, the application will consume the Google Directions API to find the fastest way to arrive from the driver´s current location to the first container in the route using real-time traffic information. When the first container is reached and served, the *next container* button is enabled to find the path from the last visited location to the next container in the route. This operation is repeated for all containers in the route. A progress bar indicates the collection progress and it is updated every time a new container is reached.
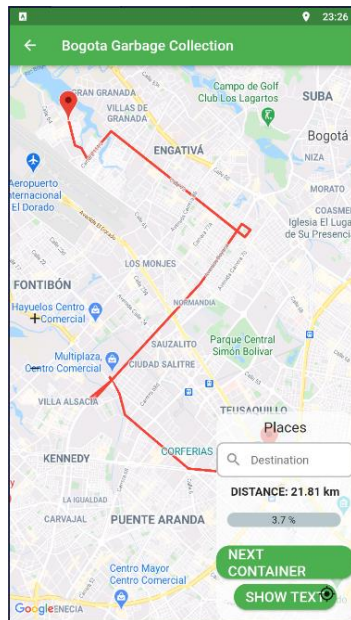
Figure 15 Route from location of the driver to the first container

5. Discussion

To evaluate the results, we will analyze them separately by layers. Regarding the data collection process, namely the "physical layer", the measurement units extract precise filling levels and communicate them efficiently to the server, and the MQTT broker acts as a bridge to connect the IoT server with Back-End services. Then, data can be stored in a database where applications can easily access them. therefore, we consider that although some elements that can be enhanced, the physical part of the system is working properly.

The logical part of the system is an approach to how to profit and manage collected data. There are possibly more precise and efficient ways to treat these data. Clustering containers need to consider terrain characteristics, hence implementing other routing strategies or using private routing services should be validated and compared with the obtained results to select the most suitable option to plan collecting operation.

It is important to mention that simulated data may not strictly match the actual waste generation in Bogotá and that they were used for academic purposes only. Therefore, this study does not compare the obtained routes against the routes that are currently being applied to collect waste in Bogotá city. Nevertheless, simulated data are valuable as they could be used as benchmark to evaluate routing algorithms performance.

There are some elements that can significantly be improved in the system. Some of them are listed below. Using LoRa as transmission technology needs some considerations. Since LoRa uses a centered star topology gateway, which is required to find an efficient gateway distribution that maximizes the coverage region by reducing number of gateways. A 50% of coverage overlapping is highly recommended so that each measure node is served by at least two gateways. This generates a failure tolerance in case any gateway is temporally out of service. Another gateway in the network might serve as back up for the nodes, thus allowing them to continue delivering measures and keeping them communicated with the server.

Gateways should be placed in altitude above the building average height, trying to generate a direct line of sight from the gateway to measurement units, so messages are delivered and received. Notwithstanding, tuning parameters such as the Spreading Factor might upgrade signal power. Therefore, when the signal is obstructed by obstacles as walls, the gateway can still receive the signal. In areas where many nodes are deployed, it is convenient to review node distribution and configuration, as suggested by (Premsankar et al., 2020) .

When grouping containers, other factors need to be reviewed, not only geographical proximity, for instance, hydric resources such as wetlands and rivers. In this work, we observed that clustering aggregates containers that were relatively close, but some of them were separated by a creek. Aggregating those containers in the same group implies to circle the river until the next bridge, which increases travel distance and affects negatively the route efficiency. To solve the problem, a clustering strategy with spatial boundaries should be implemented in such a way that elements be assigned to groups within the hydric boundaries and avoid exceeding those limits.

Before implementing the proposed collection strategy, a relocation of containers is required, hence it is necessary to find a place where containers will be easily reached by both citizens and recycling enterprises staff. This place should be situated in easy access ways, thus minimizing detours, or narrow ways where truck transit is difficult and could slow down the operation.

Measurement units need to be improved in two aspects. First, they need to manage energy efficiently. For that purpose, it is required to adapt low consumption switches that turn the devices on exclusively when they will extract measures, possibly 2 or 3 times a day, the rest of the time they should remain in sleep mode. a photovoltaic cells might be a good energy supply alternative. Second, an external case must be designed to protect circuits from weather conditions and manipulation.

Measurement units can perform well even without a GPS sensor, which is not decisive as container locations can be indexed to the data schema when units are installed and identified by a unique ID code. The containers will remain almost in the same locations, but they can be updated few times a year. This reduces the cost of measurement units in 33% approximately, but also reduces energy consumption.

To increase profits on collected material, routes should be separated. There must be at least two types of routes: one to collect general disposable waste and another for recyclable material. The latter must be properly classified and managed, avoiding it being contaminated and ending in landfills, where it becomes a source of contamination and damage for the environment.

The mobile application shows no exact alignment between the streets delivered by the maps provider and lines are used to show routes. This happens because the data used for simulation and routing employs a different geographical reference system than the maps provider. Data used for simulation and routing was coded using Magna Sirgas, the official Bogota geographical reference system, whereas the maps provider uses the World geodesic system 84 WGS84. In consequence, the paths show a slight deviation, therefore, it is necessary to find a map provider that uses a reference system similar to Magna Sirgas.

6. Conclusions

MSW has being constantly increasing in the last years, affecting human health, life style and environments. The city governments made a great effort to treat waste in a responsible and sustainable way. To tackle this situation, an efficient sensing system was proposed, which allowed cities to plan and take data driven decisions. A prototype was implemented using LoRa as transmission technology. Data flow was managed along all the network components until data was stored in a GIS database. Collected data was processed to generate collection routes that prioritize the gathering of containers reporting high levels, thus increasing the collected volume and reducing travel times. Planned routes were displayed in a mobile application in order to assist truck drivers throw the journey, as support for the operation.

This research shows why updated data are an important asset and how cities can take advantage of them to better plan typical operations, such as MSW collection. The development of this prototype demonstrates the technical viability of implementing sensing systems enabled by IoT and GIS in real environments. It also shows that logistic processes, such as MSW collection, can be enhanced with communication technologies to be more efficient, in this particular case by supplying the collecting enterprises with a support application, while reducing container saturation and overflow.

For a real environment deployment some points need to be analyzed: i) sensors must be adjusted with a directive angle to acquire accurate measurements; ii) efficiently managing energy on measurement devices; iii), the spreading factor configuration when there are many nodes in a small area; iv) the gateway location in order to generate a direct line of sight between gateway and sensing units; and v) including relevant and detailed terrain characteristics so that routes can be computed more precisely.

For future work, a comparison of the proposed routes with the currently used routes can be used to validate if the proposed system has a valuable impact on the collecting operation. Collection routes to retrieve reusable material should be independent from routes to recover the non-recyclable one. When adjusted, the model could be extended and generalized as an open framework to manage MSW in an efficient and sustainable way.

Code Availability

Design and code have being published in a Github repository for public use and contribution available at https://github.com/montanarco/WSNGarbageRouting

REFERENCES

Abdulkader, M. M. S., Gajpal, Y., & Elmekkawy, T. Y. (2015). Hybridized ant colony algorithm for the Multi Compartment Vehicle Routing Problem. *Applied Soft Computing Journal*, *37*, 196–203. https://doi.org/10.1016/j.asoc.2015.08.020

Amal, L., Son, L. H., & Chabchoub, H. (2018). SGA: spatial GIS-based genetic algorithm for route optimization of municipal solid waste collection. *Environmental Science and Pollution Research*, *25*(27), 27569–27582.

https://doi.org/10.1007/s11356-018-2826-0

Babaee Tirkolaee, E., Abbasian, P., Soltani, M., & Ghaffarian, S. A. (2019). Developing an applied algorithm for multi-trip vehicle routing problem with time windows in urban waste collection: A case study. *Waste Management and Research*, *37*(1_suppl), 4–13. https://doi.org/10.1177/0734242X18807001

Bingol, E., Kuzlu, M., & Pipattanasompom, M. (2019). A LoRa-based Smart Streetlighting System for Smart Cities. *7th International Istanbul Smart Grids and Cities Congress and Fair, ICSG 2019 - Proceedings*, 66–70. https://doi.org/10.1109/SGCF.2019.8782413

Boysen, J. (2017, August 25). Austin Waste and Diversion. Kaggle Data Set. https://www.kaggle.com/jboysen/austin-waste

Bustos Coral, D., Oliveira Santos, M., Toledo, C., & Fernando Niño, L. (2018). Clustering-Based Search in a Memetic Algorithm for the Vehicle Routing Problem with Time Windows. *2018 IEEE Congress on Evolutionary Computation, CEC 2018 - Proceedings*, 8. https://doi.org/10.1109/CEC.2018.8477710

Cambier, R. (2016, September 13). TheThingsNetwork/java-app-sdk. TTN-Java-App-Sdk. https://github.com/TheThingsNetwork/java-app-sdk

Campbell, A. M., & Wilson, J. H. (2014). Forty years of periodic vehicle routing. *Networks*, *63*(1), 2–15. https://doi.org/10.1002/net.21527

Cerchecci, M., Luti, F., Mecocci, A., Parrino, S., Peruzzi, G., & Pozzebon, A. (2018). A Low Power IoT Sensor Node Architecture for Waste Management Within Smart Cities Context. *Sensors*, *18*(4), 1282. https://doi.org/10.3390/s18041282

Chaudhari, S. S., & Bhole, V. Y. (2018). Solid Waste Collection as a Service using IoT-Solution for Smart Cities. *2018 International Conference on Smart City and Emerging Technology, ICSCET 2018*, 1–5. https://doi.org/10.1109/ICSCET.2018.8537326

Chaudhary, S., Nidhi, C., & Rawal, N. R. (2019). Gis-based model for optimal collection and transportation system for solid waste in allahabad city. *Advances in Intelligent Systems and Computing*, *814*, 45–65. https://doi.org/10.1007/978-981-13-1501-5_5

Coupey, J. (2018, August 17). Solving vehicle routing problems with OpenStreetMap and VROOM. Stateofthemap. https://2018.stateofthemap.org/2018/ T053-Solving_vehicle_routing_problems_with_OpenStreetMap_and_VROOM/

De Castro Tomé, M., Nardelli, P. H. J., & Alves, H. (2019). Long-Range Low-Power Wireless Networks and Sampling Strategies in Electricity Metering. *IEEE Transactions on Industrial Electronics*, *66*(2), 1629–1637. https://doi.org/10.1109/TIE.2018.2816006

Faulin, J., Juan, A., Lera, F., & Grasman, S. (2011). Solving the capacitated vehicle routing problem with environmental criteria based on real estimations in road transportation: A case study. *Procedia - Social and Behavioral Sciences*, *20*, 323–334. https://doi.org/10.1016/j.sbspro.2011.08.038

Fujdiak, R., Masek, P., Mlynek, P., Misurec, J., & Olshannikova, E. (2016). Using genetic algorithm for advanced municipal waste collection in Smart City. 2016 10th International Symposium on Communication Systems, Networks and Digital Signal Processing, CSNDSP 2016. https://doi.org/10.1109/CSNDSP.2016.7574016

Gilardino, A., Rojas, J., Mattos, H., Larrea-Gallegos, G., & Vázquez-Rowe, I. (2017). Combining operational research and Life Cycle Assessment to optimize municipal solid waste collection in a district in Lima (Peru). Journal of Cleaner Production, 156, 589–603. https://doi.org/10.1016/j.jclepro.2017.04.005

Gouveia, L., & Martins, P. (1999). The Capacitated Minimal Spanning Tree Problem: An experiment with a hop-indexed model. Annals of Operations Research, 86(0), 271–294. https://doi.org/10.1023/a:1018911003529

Heidelberg University. (1997, February 19). TSPLIB. TSPLIB. http://elib.zib.de/pub/mp-testdata/tsp/tsplib/tsplib.html

Jwad, Z. A., & Hasson, S. T. (2018). An Optimization Approach for Waste Collection Routes Based on GIS in Hillah-Iraq. ICOASE 2018 - International Conference on Advanced Science and Engineering, 60–63. https://doi.org/10.1109/ICOASE.2018.8548889

Kang, K. D., Kang, H., Ilankoon, I. M. S. K., & Chong, C. Y. (2020). Electronic waste collection systems using Internet of Things (IoT): Household electronic waste management in Malaysia. Journal of Cleaner Production, 252. https://doi.org/10.1016/j.jclepro.2019.119801

Karthikeyan, S., Rani, G. S., Sridevi, M., & Bhuvaneswari, P. T. V. (2018). IoT enabled waste management system using ZigBee network. RTEICT 2017 - 2nd IEEE International Conference on Recent Trends in Electronics, Information and Communication Technology, Proceedings, 2018-Janua, 2182–2187. https://doi.org/10.1109/RTEICT.2017.8256987

Kaza, S., Yao, L. C., Bhada-Tata, P., & Van Woerden, F. (2021). What a Waste 2.0 : A Global Snapshot of Solid Waste Management to 2050. World Bank Group.

Kooijman, M. (2020, October 3). Arduino-lmic. GitHub. https://github.com/matthijskooijman/arduino-lmic

Kuo, R. J., & Zulvia, F. E. (2017). Hybrid genetic ant colony optimization algorithm for capacitated vehicle routing problem with fuzzy demand - A case study on

garbage collection system. 2017 4th International Conference on Industrial Engineering and Applications, ICIEA 2017, 244–248. https://doi.org/10.1109/IEA.2017.7939215

Li, Q., Liu, Z., & Xiao, J. (2018). A Data Collection Collar for Vital Signs of Cows on the Grassland Based on LoRa. Proceedings - 2018 IEEE 15th International Conference on e-Business Engineering, ICEBE 2018, 213–217. https://doi.org/10.1109/ICEBE.2018.00041

Lozano, Á., Caridad, J., De Paz, J. F., González, G. V., & Bajo, J. (2018). Smart waste collection system with low consumption LoRaWAN nodes and route optimization. Sensors (Switzerland), 18(5). https://doi.org/10.3390/s18051465

Medvedev, A., Fedchenkov, P., Zaslavsky, A., Anagnostopoulos, T., & Khoruzhnikov, S. (2015). Waste management as an IoT-enabled service in smart cities. Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 9247, 104–115. https://doi.org/10.1007/978-3-319-23126-6_10

Meesala, Sirisha & Manoj Kumar, Nallapaneni & Parimala, Sugathi & Jyothi, N. Arun. (2018). Monitoring the smart garbage bin filling status: An IoT application towards waste management. International Journal of Civil Engineering and Technology. 9. 373-381.

Nazeer, K. A & Sebastian, M.. (2009). Improving the Accuracy and Efficiency of the k-means Clustering Algorithm. Lecture Notes in Engineering and Computer Science. 2176.

Ospina-Pinto, C., Rincón-Pardo, M., Soler-Tovar, D., & Hernández-Rodríguez, P. (2017). The role of rodents in the transmission of Leptospira spp. In swine farms. Revista de Salud Publica, 19(4), 555–561. https://doi.org/10.15446/rsap.v19n4.41626

Papalitsas, C., Karakostas, P., Andronikos, T., Sioutas, S., & Giannakis, K. (2018). Combinatorial GVNS (General Variable Neighborhood Search) Optimization for Dynamic Garbage Collection. Algorithms, 11(4), 38. https://doi.org/10.3390/a11040038

Pardini, K., Rodrigues, J. J. P. C., Diallo, O., Das, A. K., de Albuquerque, V. H. C., & Kozlov, S. A. (2020). A Smart Waste Management Solution Geared towards Citizens. Sensors, 20(8), 2380. https://doi.org/10.3390/s20082380

Premsankar, G., Ghaddar, B., Slabicki, M., & Francesco, M. Di. (2020). Optimal Configuration of LoRa Networks in Smart Cities. IEEE Transactions on Industrial Informatics, 16(12), 7243–7254. https://doi.org/10.1109/TII.2020.2967123

Sadowski, S., & Spachos, P. (2020). Wireless technologies for smart agricultural monitoring using internet of things devices with energy harvesting capabilities. Computers and Electronics in Agriculture, 172, 105338. https://doi.org/10.1016/j.compag.2020.105338

Sagir, S., Kaya, I., Sisman, C., Baltaci, Y., & Unal, S. (2019). Evaluation of Low-Power Long Distance Radio Communication in Urban Areas: LoRa and Impact of Spreading Factor. Proceedings - 2019 7th International Conference on Digital Information Processing and Communications, ICDIPC 2019, 68–71. https://doi.org/10.1109/ICDIPC.2019.8723666

Sharma, A. (2016, September 25). VROOM-Project. Vehicle Routing Open-Source Optimization Machine. https://github.com/VROOM-Project/vroom/blob/master/docs/API.md

Vu, D. D., & Kaddoum, G. (2017). A waste city management system for smart cities applications. *Proceedings - 2017 Advances in Wireless and Optical Communications, RTUWO 2017*, 2017-January, 225–229. https://doi.org/10.1109/RTUWO.2017.8228538

Wu, B. Y., Lancia, G., Bafna, V., Chao, K. M., Ravi, R., & Tang, C. Y. (2000). A polynomial-time approximation scheme for minimum routing cost spanning trees. *SIAM Journal on Computing*, *29*(3), 761–778. https://doi.org/10.1137/S009753979732253X

Yusof, N. M., Zulkifli, M. F., Mohd Yusof, N. Y. A., & Azman, A. A. (2018). Smart waste bin with real-time monitoring system. *International Journal of Engineering and Technology(UAE)*, *7*(2.29 Special Issue 29), 725–729. https://doi.org/10.14419/ijet.v7i2.29.14006

Zguira, Y., Rivano, H., & Meddeb, A. (2018). Internet of Bikes: A DTN Protocol with Data Aggregation for Urban Data Collection. *Sensors*, *18*(9), 2819. https://doi.org/10.3390/s18092819

Zhao, W., Lin, S., Han, J., Xu, R., & Hou, L. (2018). Design and Implementation of Smart Irrigation System Based on LoRa. *2017 IEEE Globecom Workshops, GC Wkshps 2017 - Proceedings*, 2018-January, 1–6. https://doi.org/10.1109/GLOCOMW.2017.8269115

Zhou, Qing & Benlic, Una & Wu, Qinghua & Hao, Jin-Kao. (2018). Heuristic search to the capacitated clustering problem. European Journal of Operational Research. 273. https://10.1016/j.ejor.2018.08.043.