A WSN-based monitoring system to control Sewerage

Javier Rocher, José Luis García-Navas, Oscar Romero, Jaime Lloret
Instituto de Investigación para la Gestión Integrada de zonas Costeras.
Universitat Politècnica de València Valencia,Spain
jarocmo@doctor.upv.es, jogarna3@teleco.upv.es, oromero@dcom.upv.es, jlloret@dcom.upv.es

***Abstract*— The sewerage is a critical infrastructure in cities because of the drainage of the urban runoff and the evacuation of the wastewater. Two types of sewerage, separated sewerage and combined sewerage, can be differentiated. In this paper, we show the application of a level sensor and a rain sensor for monitoring the separated sewerage. The level sensor is used for knowing if there is a critical level of water in the sewerage. The rain sensor is used to know if it is raining. The combination of this information allows the identification of three scenarios. These scenarios are normal situation, low drainage and illicit discharge/blockages in the pipeline. In addition, we study the use of sensors and mathematical models for monitoring the velocity of the wastewater. We concluded that the use of mathematical models is a good option for monitoring the velocity. Because with exception of the thermal sensors the other types of sensors show important gaps. The velocity is used to estimate the flow that is dumping in the water bodies. We use an ESP32 board program with Arduino IDE for data collection and sending the data to a server on the same network via Wi-Fi. The server is a computer that processes the data. We present the programming code and the ports that should be used for transmitting the data from Arduino to computer server.**

Keywords—WSN, Wastewater, Level sensors, Rain Sensors.

I. Introduction

The sewerage networks are the infrastructure that allows the transport of the residual water to wastewater treatment plants (WWTP) and prevents flooding by rainwater in urban areas. The use of sewerage started in 3750 AC. However, the first modern sewerage was designed in Hamburgo in 1842. The use of these infrastructures has been demonstrated to be a measure for reducing the mortality of the population [1]. We can differentiate between two types of sewerage, separated or combined sewerage [2-3]. The difference between these two systems is the drainage of the rainwater. In the combined system the runoff of rainwater (in urban area) is collected in the same pipes as the wastewater. This water will go to a WWTP where it will be treated. In the separated systems, the wastewater and runoff go in separate pipes. In addition, in some sewerage-separated systems, the domestic wastewater and industrial wastewater have different pipes. The wastewater will go to a WWTP and the runoff will be poured into a body of water (river, lake, sea, etc.).

The combined sewerage is typical of the centre of the cities by inheritance from the constructive methods of past eras. In the new areas of urban expansion, the separate sewage system is more common. The construction of separated sewerage is more expensive than combined sewerage because it is necessary to place two pipes. In separated sewerage, the rainwater is not treated in WWTP, which reduces the maintenance cost.

Regardless of the type of sewerage, some problems are: broken pipes, the runoff rainwater, lack of power or failure in pumps and the pipeline blockage [4]. The sewerage networks have overflow channels in different points of its distribution. These are located in an upper bound of the pipeline. These overflow channels are used to transport the excess water. These situations will happen when the sewerage is blocked or too much water enters (storm events). In storm events in separate sewerage, the wastewater can enter into the storm sewerage, if combined sewerage is connected to separate sewerage. Xu et al. studied the non-stormwater in storm drains, in the city of Shangai. They concluded that 51% of water was an inappropriate entry of sewage flow into the storm drains [5]. However, different studies have been showing high contamination levels in the runoff rainwater in the separate sewerage [6]. In addition, the presence of illicit connection or wastewater pipes that discharged in the rainwater is hazardous of water bodies because the water is not treated. WSN have been developed in the last years for monitoring the environment, smart cities, e-health, etc. Different architectures have been developed in smart cities [7], farming [8], industry [9], health [10], etc.

In this paper, we propose a system based on level sensors and rain sensors for monitoring the water level in separated sewerage. In addition, we study the application of sensors and mathematical models to measure the water velocity in the pipes. The sensors are connected to microcontroller ESP32 module to send the data to a computer server. The computer server will treat the data to generate the different alarms of the three possible scenarios. The level sensor is located in the trap of sewerage to monitor the level of water. When rain sensors and level sensors have a negative answer, the sewerage is ok. This will also happen when water is detected in the sewer and rain, that will indicate that rainwater is being drained. When the rain sensor does not detect rain, but the level sensor detects water the system sends an alarm. Because if the system detects rain but does not detect water in the sewerage this is an indicator of problems in the drainage system.

The rest of the paper is structured as follows: Section II shows the related works. In section III, we study the different sensors and mathematical models for monitoring the water velocity in the pipes. In the section IV, we show the algorithm of decision. In the Section V we show the programming that we use for data recollecting and sending it to a server. In section VI we show the simulation of the different scenarios of our system Finally, in section VII we presented the conclusions of the work and future works.

II. Related Work

In this section we analyze different studies related to monitoring the sewerage.

Rocher et al. [11] showed an inductive sensor for monitoring the water level in a pipe. The sensor was based on two coils. A coil is powered by a discontinuous current and the other coil is induced. This system has been tested in pipelines of 32 mm. See et al. [12] showed a system based on ultrasonic level sensors for monitoring the blockage in sewerage. The system detects if the level of water in the pipeline is high, normal or low. If the water level is high the sensor node sends an alarm. They used Zigbee connection for saving energy and every node is connected to a data gatherer, for managing the data. These systems are used for measuring the water level in the pipes just as we proposed. However, they do not take into account the rain. For this reason, in a storm event, these systems will generate a false alarm.

Khulief et al. [13] studied the use of two hydrophones (B&K PULSE type 3560-B) for measuring the presence of leaks in pipes. They used the hydrophones for data acquisition and a PC for processing the data. The leaks generate a series of acoustic waves that can be measured with the hydrophones. The system has to differentiate between the acoustic waves generated by a leak and the other wave sources.

Irvine et al. [14] studied different parameters for measuring that have a low cost in a laboratory to detect clandestine waste discharge. They investigated in the sewerage of Western New York State. First, they performed a visual reconnaissance of the storm sewerage system. They tested different parameters for detecting illicit wastewater and determining the best parameters for tracking the illicit wastewater dumps. These parameters are pH, conductivity, Escherichia *coli* (microorganism), ammonia, nitrate, fluoride, total chlorine, potassium, detergents, phosphorous, and, turbidity. They were used for detecting illicit discharges especially the presence of *Escherichia coli*. Hauser et al. [15] have used pH, conductivity and chloride electrodes for detecting clandestine waste discharges. In these articles different parameter values were used for detecting the illicit discharge. The problem is the cost of monitoring these parameters. Our system cannot analyze the value of the pollutants. However, we think that it can detect the discharge and that the responsible personnel can take a water sample.

Talukder et al. [16] presented a system for preventing the blockage in pipes. They used an arduino as a microcontroller, NRFmodule, acoustic sensors and a servo motor along with its power supply. The system is located on the pipe with a lifting platform that filtered the water and the materials that can block the pipe. When the acoustic sensor detects an accumulation of waste material the system starts with the servo motor the platform rises and pours the contents out of the pipe. Dhananchezhiyan et al. [17] developed a robot for removal of blockage in the sewerage. The robot uses an infrared proximity sensor for detecting the presence of blockage in the pipe. The robot has a manipulator disc for removing the blockage. These systems are designed for the elimination of blockage in the pipelines. However, they cannot be placed continuously in a pipe because its moving parts would be blocked in a short time.

Nienhuis et al. [18] proposed the use of Distributed temperature sensing with fiber-optic cable for detecting illicit connection in storm sewerage. The system is based on the difference of temperature between the stormwater and the illicit water. In this case, the illicit discharge can be identified, however, they cannot detect the presence of a blockage. In addition, in the event of a storm the temperature of the water may be different according to its route, which causes false positives.

In this paper, we present a system for monitoring the water level and velocity of the water in the pipes. The different acoustic systems presented do not take into account the rain. The other systems presented systems to eliminate the blockage in pipelines. However, these systems have moving parts that is not recommended in sewerage. Finally, different authors propose the use of different parameters for monitoring the illicit discharge. In these cases, there are many parameters that cannot use online monitoring and the system present high prices.

III. VELOCITY MONITORING.

In this section, we show the different technologies that can be used for monitoring the speed of water in sewerage.

Different methods can be used for monitoring the flow in sewerage. Acoustic, magnetic, oscillatory, turbine, thermal, Coriolis, etc. [19] In addition, different authors have proposed mathematical models for modelling the water velocity for pipe design. These models can be used for modelling the water velocity.

The sewage has many solids that can block the moving parts of the sensors. For this reason, any sensor with moving parts is not recommended for monitoring the sewerage. The magnetic, ultrasonic, thermal and Coriolis methods are not contacted in water for this reason they can be used in the sewerage. However, these sensors present important gaps that prevent their use in sewerage. The magnetic flow sensors are based on the Faraday law. These sensors can only be used in full pipes. The sewerage is not full of water under normal conditions. Acoustic methods use the velocity of ultrasonic waves to measure the velocity of the water. Sewerage has important changes in the solids transported, bubbles and level of water in the pipe. These changes can negatively affect the measures. Finally, the Coriolis sensors are based on the conservation of the angular moment. These sensors present a gap with large size.

Thermal sensors are the best option for monitoring the water velocity in a pipe. However, these sensors are commonly used in airflow monitoring. There are three methods, detecting a pick of temperature downstream, detecting the energy needed to maintain a filament at a constant temperature or measuring the temperature change between upstream and downstream. Another solution is the use of hydrodynamic models for monitoring the water velocity. The Manning equation (equation 1) is a solution. This equation can be used in open channels or pipes that are not full of water. The values of manning rugosity can be found in the bibliography and the pending is a value of the pipe infrastructure.

$$V = \frac{1}{n} * Rh^{\frac{2}{3}} * \sqrt{S} \qquad (1)$$

Where:

V: is the velocity (m/S)
n: is the manning rugosity
Rh: is the hydraulic radius
S is the pending (m/m)

Another equation is the Hazen-Williams equation. This is used when the pipe is full of water and it is shown in the equation 2.

$$V = 0.8494 * C * Rh^{0.63} * S^{0.54} \qquad (2)$$

Where:

C: is the coefficient of rugosity

We should study which method is the best for monitoring the velocity of water in sewerage. The use of a thermal sensor has the advantage that it is a real measure of the environment. However, energy consumption is very large. Instead, the use of mathematical models do not need the use of extra energy, however, we do not measure the real velocity.

IV. ALGORITHM.

In this section, we show the algorithm of decision of our system.

The algorithm is represented in Fig. 1. In the first place, the water level sensor measures the level of water in the sewerage. The system compares the water level now (Wn) with a water level critical (Wc). The Wc is a level defined by the user for indicating from which point the water level is not in a normal situation. After, the system checks if it is raining or not. Finally, the system compares the water level in a point (Wu) with the water level of the next control point downstream (Wd). This algorithm is used when the pipes have the same diameter. In the case of the pipe presented, we have different diameter between two points so we use the velocity. The velocity can be calculated with the water level and the knowledge of the characteristics of the pipes. The algorithm will be similar, the only change being the water quantity comparisons by the speed.

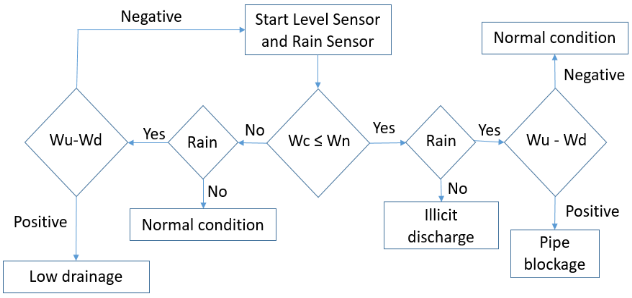

Fig. 1. Algorithm for data processing

The definition of a Wc is important for discharged false positive. The value of Wc should be determined by the person responsible in order to prevent small amounts of water arising from the irrigation of gardens, private vehicles, etc. setting of alarms that are not related to the illicit dumping.

V. WI-FI MODULE AND SERVER PROGRAMMING

In this section, we are going to present the design of a Wi-Fi module used to receive information from the sensor explained before and to transmit this information via Wi-Fi to a server located in a Computer connected to the same Network.

The module that has been used is the ESP32 Board, as can be seen in Fig. 2. The ESP32 Board was programmed with Arduino IDE.

This module permits to connect to him such number of inputs as DX Pins it has, where X represent the number of the Pin. It also has Voltage pin (3V3) and a GND pin, that are necessary to connect sensors to positive and negative voltage. Finally, each sensor must be connected to Arduino ESP32 Board by one of the DX pins explained before.

The module must be able to stablish a connection to a Wi-Fi network. Other authors use Wi-Fi network for monitoring the sewerage. Once the module is connected, it starts asking for information to every sensor it has connected. The information received is captured and sent to a server programmed and activated previously. The way the information is sent is by a HTTP Post request sent to the IP direction and Port where the server is listening to.

Once the information is received on the server it can be processed and the algorithm explained before can be used to determinate if the sensor values obtained determinates some risk situations according to the sewerage networks.

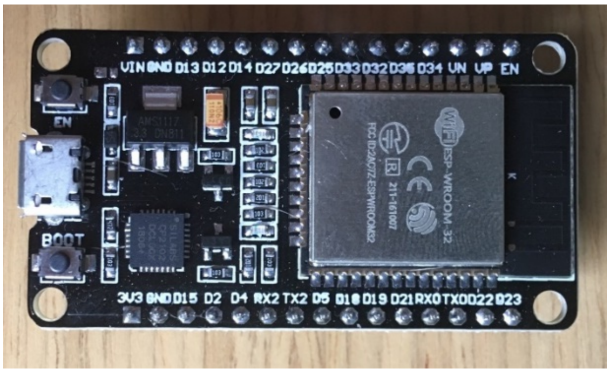

Fig. 2. ESP32 Board

In the next paragraphs we are going to explain how the server and the ESP32 have been programmed and what are the most important parts of the codes that are necessary to the study.

A. *Server*

In order to program the server that is going to receive the information we have used Express, an application infrastructure of Node.js. As we are going to receive the information using a HTTP Post, we need to use the post function that has two variables. The first one is the URL direction and the second one is the function wanted to be executed when a Post Request is received. Inside the brackets of this function is where the code needs to be added. This function has also two variables: *req* contains all the information of the post, and using *req.body.variable1* it is possible to access to the variable with name *variable1*; *res* is used to respond to the request.

Once the post is programmed in the server it is necessary to open the server selecting the port number and the IP direction and using the function *listen(port, ip).*

Fig. 3 shows how to program all the functions explained before. In order to initiate the server, it is necessary to use the Command Line Window, navigate to the folder that contains the file server.js and execute “node server.js”. The server will be listening in http://192.168.1.39:8080.

```
app.post('/sensorData', function(req,res){
});
var port = 8080;
var ip = "192.168.1.39";
app.listen(port,ip);
```

Fig. 3. Server HTTP Post Function

B. ESP32 module

The ESP32 module needs different blocks of code for each functionality we want to implement.

In order to connect the module to Wi-Fi, first of all it is necessary to include the library, then it is needed to declare the ssid of the Wi-Fi network and the password of the connection and finally connection can start. It is shown in Fig. 4.

```
#include <WiFi.h>
const char* ssid = "networkName "
const char* password = "password";
WiFi.begin(ssid, password);
```

Fig. 4. Wi-Fi programing on ESP32

It is also possible to probe if the connection is available or if unavailable, it turns off with the comparison shown in Fig. 5.

```
WiFi.status() == WL_CONNECTED
```

Fig. 5. Comparison to probe Wi-Fi connection

In order to send the information via HTTP Post Request it is necessary to include the library HTTPClient and then use it with the functions of Fig. 6.

```
#include <HTTPClient.h>
HTTPClient http;
http.begin("http://192.168.1.39:8080
/sensorData");
http.addHeader("Content-type","application/
x-www-form-urlencoded");
int respuesta = http.POST(vars);
```

Fig. 6. HTTP Post Request Programing

As shown in lines before, the HTTPClient must be initiated with the function *begin* indicating the URL where the server is listening. After the IP direction and the port, it is necessary to write the URL name of the post function. In that it is "sensorData".

Also, it is essential to add a Header to the Post Request establishing the content-type of the request, in that case it is an **x-www-form-urlencoded.** For that reason, when doing the *http.POST(vars)*, the variable *vars* (string) must be as follows:

"var1=value1&var2=value2&…."

Where each variable must be separated from its corresponding value with an "=" and each pair variable-value must be separated from others by "&".

In the case of sensors, each one could need or not the definition of its library and for each variable there is a different way to take its corresponding value.

For example, in case of the rain sensor FC-37 it doesn't need any library to be #define. The way the information from this sensor can be taken in declaring the pin the sensor is going to be connected to and reading its value. FC-37 permits to take analogical or digital response. In this study is more relevant a digital response such a Boolean variable.

```
boolean RAIN= false;
int pinRain = 5;
pinMode(pinRain, INPUT);
RAIN=!(digitalRead(pinRain));
if(RAIN){ Serial.println("It is Raining");
}else{ Serial.println("It is not Raining ");
}
```

Fig. 7. Rain Sensor Programing

Fig. 7 presents all declarations and functions needed to program the FC-37 sensor. First of all, two variables are defined. RAIN variable that contains information about if it is raining or not and pinRain that contains the number of the physical pin the sensor is connected to. Then, the function pinMode determinates that the pin is used as an input. Finally, with the function digitalRead(pinRain) the values of the sensor can be read, and with the if-else can be confirmed whether it is raining or not.

Except the code lines about Wi-Fi connection and those that are used to define libraries or variables, the rest of the code lines, that refers to taking sensors data and transmitting them, are written inside of a *loop(),* that means that they are executed continually. It is possible to determine how much time passes between each execution, or what is the same, how often the sensor data is taken. In order to determinate that time, *delay(int delay)* function can be used. This function must have an *int* variable that is time in ms.

For those sensors whose value, sent to the ESP32 module, is a float and it is necessary to transmit it via the HTTP Post, it is essential to parse it to type string. To do it we have used the code presented in Fig. 8.

```
String floatToString( float n, int l, int d,
boolean z){
    char c[l+1];
    String s;
    dtostrf(n,l,d,c);
    s=String(c);
    if(z){  s.replace(" ","0"); }
    return s;
}
```

Fig. 8. floatToString

C. Structure of the system

In this section, we show the Structure of the system proposed.

We have two sensors (acoustic sensor level and rain sensor). The Water level sensor is locating below of the sewer cover. This location allows the sensor to measure the water level in the sewerage and allows for easy manipulation in the case that the sensor node need reparation or maintenance.

The rain sensor is located on the surface. We need to find a location that does not have buildings nearby. The buildings can stop part of the rain causing underestimates of the rain. Examples of these locations are roofs of the high buildings, roundabouts, garden areas, etc. The two sensors send the information to a server, where the algorithm is executed and sends the alarms.

In the Fig. 9 we can observed an example of the localization of the proposed system.

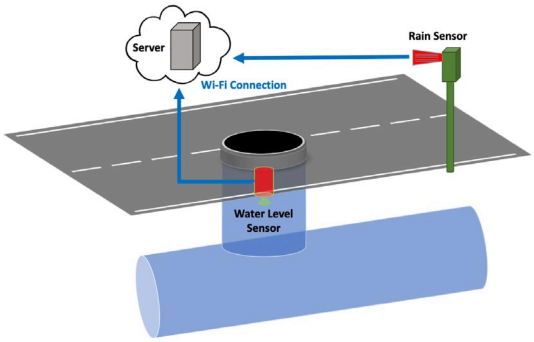

Fig. 9. Location of proposed system

VI. Results

In this section, we study the different scenarios that can be analyzed in the section IV. These different scenarios presented are simulated. We simulate the behaviour of a stretch of sewerage that have two acoustic sensors at its ends when rain starts and when it finishes.

The first scenario is the normal condition. This condition happens in two situations. When it is not raining and the sewerage does not transport water. The other normal situation is when it is raining and the sewerage has water. This situation is represented in Fig. 10. In this situation, the rain starts at 25 minutes and finishes at 150 minutes. The two water level sensors are located upstream and downstream. The downstream will register a value of water level higher because at this point more water will accumulate. If we subtract the height value of the two sensors, we will obtain a difference that will be less than what we will obtain in the other cases.

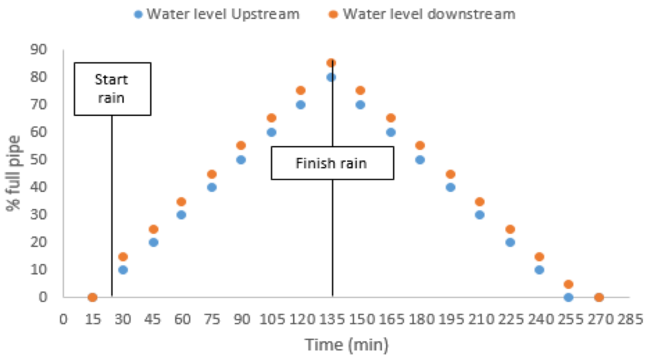

Fig. 10. Normal scenario

Another scenario is when the pipe is stuck. In this situation, the rainwater is accumulated in the sewerage during the rain before the blockage. When this happens there is much more water in the upper area compared to the lower area because of this the drainage will occur very slowly. This situation is represented in the Fig. 11. In this case, an unlocking device should be sent to the pipe in order to eliminate the jam.

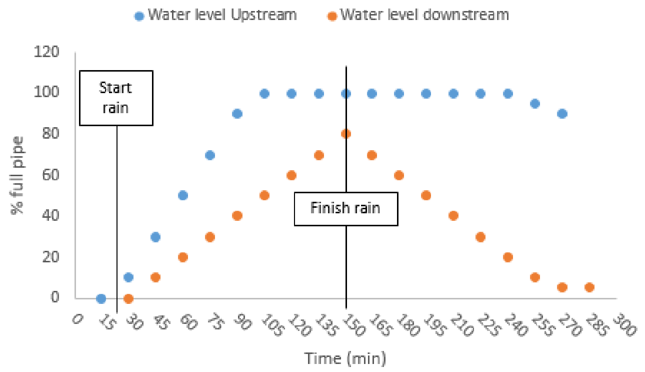

Fig. 91. Blockage scenario

When a problem in the drainage of the runoff occurs, we can observe that the value of the water level upstream is much smaller than the value of the water level downstream. This occurs because upstream water is not entering the sewer while downstream will be entering. In this case the system generates an alarm so that drains are monitored at street level. If the problem is not solved in a short space of time floods can occur in lower areas of the city. In Fig. 12, it is represented this situation.

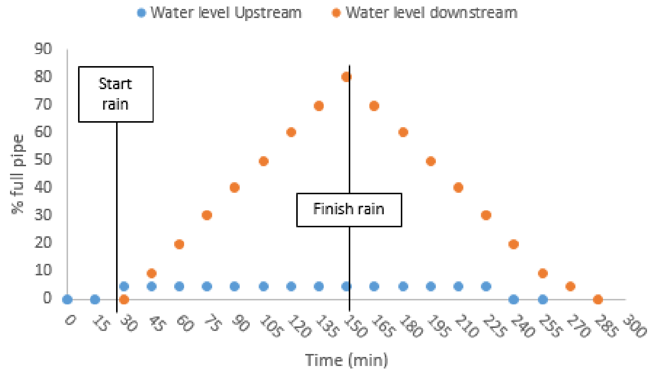

Fig. 102. Low drainage

Finally, in the case we detect there is no rain and water level is larger than the critical water level this is a sign of illicit discharge. In this case, we assume that the discharge is made somewhere between the two measurement points. The critical water level is defined by the user and it is used to discard possible positives due to the irrigation of parks, washing of private vehicles, etc. that use water and could end up in the sewer system. This situation is represented in the Fig. 13.

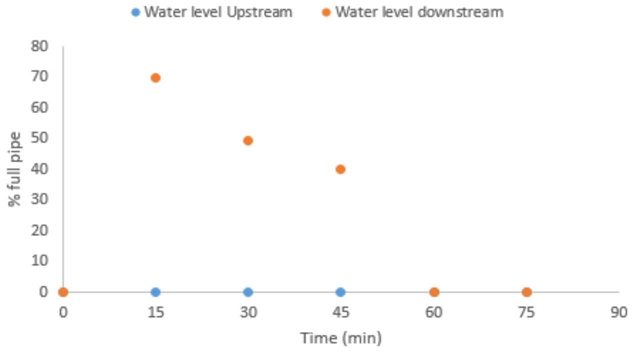

Fig. 113. Illicit discharge

VII. Conclusions and Future works.

In this paper, we presented a new system for the monitoring of the water level in the sewerage. The system is composed of two sensors: a level sensor and a rain sensor. The level sensor is used for monitoring the water level in the sewerage in order to determine the speed of the water indirectly. With the knowledge of the amount of water and the presence of rain, it is possible to determine 3 scenarios, a normal situation, a lack of drainage and the presence of an illegal spill or blockage in the pipeline. We use an ESP32 board as the microcontroller of the sensors in order to send the data to a server.

In future works, we would test the system in real conditions and add more sensors in our system as turbidity or conductivity. In addition, we would add level sensors in the pipes of wastewater in a separated system for monitoring the residual water that reaches the rainwater pipes. For this purpose, it will be important to have a secure wireless network [20] between the different sensor nodes and with Energy harvesting protocols as the presented in [21]. In addition, in the future we will work in the development an algorithm for energy saving as the presented in [22]

Acknowledgment

This work has been partially supported by the European Union through the ERANETMED (Euromediterranean Cooperation through ERANET joint activities and beyond) project ERANETMED3-227 SMARTWATIR by the "Ministerio de Educación, Cultura y Deporte", through the "Ayudas para contratacion predoctoral de Formación del Profesorado Universitario FPU (Convocatoria 2016)". Grant number FPU16/05540.

References

[1] M. Alsan and C. Goldin, "Watersheds in Child Mortality: The Role of Effective Water and Sewerage Infrastructure, 1880–1920," Journal of Political Economy, vol. 127, no. 2, pp. 586–638, Apr. 2019

[2] G. Mannina and G. Viviani, "Separate and combined sewer systems: a long-term modelling approach," *Water Science and Technology*, vol. 60, no. 3, pp. 555–565, Jul. 2009.

[3] S. Zgheib, R. Moilleron, and G. Chebbo, "Priority pollutants in urban stormwater: Part 1 – Case of separate storm sewers," Water Research, vol. 46, no. 20, pp. 6683–6692, Dec. 2012.

[4] Environmental Protection Agency (EPA), "Licensing Guidelines for Sewage Treatment Systems," EPA, Jul. 2013.

[5] Z. Xu, H. Yin, and H. Li, "Quantification of non-stormwater flow entries into storm drains using a water balance approach," Science of The Total Environment, vol. 487, pp. 381–388, Jul. 2014.

[6] P. . Mikkelsen, "Retrofitting Urban Drainage Systems Using Best Stormwater Management Practices — Some Scandinavian Experiences," in Enhancing urban environment by environmental upgrading and restoration, Dordrecht: Springer, 2005, pp. 1–12

[7] L. Parra, J. Rocher, S. Sendra, and J. Lloret, "An Energy-Efficient IoT Group-Based Architecture for Smart Cities," in Energy Conservation for IoT Devices Concepts, Paradigms and Solutions, Singapore: Springer, 2019, pp. 111–127.

[8] L. Parra et al., "Design of a WSN for smart irrigation in citrus plots with fault-tolerance and energy-saving algorithms," Network Protocols and Algorithms, vol. 10, no. 2, p. 95, Jun. 2018.

[9] G. Zhao, "Wireless Sensor Networks for Industrial Process Monitoring and Control: A Survey," *Network Protocols and Algorithms*, vol. 3, no. 1, Apr. 2011.

[10] M. Atto and C. Guy, "Routing Protocols for Structural Health Monitoring of Bridges Using Wireless Sensor Networks," *Network Protocols and Algorithms*, vol. 7, no. 1, pp. 1–23, Apr. 2015.

[11] J. Rocher, L. Parra, J. Lloret, and J. Mengual, "An Inductive Sensor for Water Level Monitoring in Tubes for Water Grids," in 2018 IEEE/ACS 15th International Conference on Computer Systems and Applications (AICCSA), 2018.

[12] C. See, K. Horoshenkov, R. abd-alhmeed, Y. Hu, and S. Tait, "A Low Power Wireless Sensor Network for Gully Pot Monitoring in Urban Catchments," IEEE Sensors Journal, vol. 12, no. 5, 2011.

[13] Y. A. Khulief, A. Khalifa, R. B. Mansour, and M. A. Habib, "Acoustic Detection of Leaks in Water Pipelines Using Measurements inside Pipe," Journal of Pipeline Systems Engineering and Practice, vol. 3, no. 2, pp. 47–54, May 2012.

[14] K. Irvine, M. C. Rossi, S. Vermette, J. Bakert, and K. Kleinfelder, "Illicit discharge detection and elimination: Low cost options for source identification and trackdown in stormwater systems," Urban Water Journal, vol. 8, no. 6, pp. 379–395, Nov. 2011.

[15] F. M. Hauser, T. Metzner, T. Rößler, M. Pütz, and S. Krause, "Real-time wastewater monitoring as tool to detect clandestine waste discharges into the sewage system," Environmental Forensics, vol. 20, no. 1, pp. 13–25, Jan. 2019.

[16] S. Talukder, M. I. I. Sakib, Z. R. Talukder, U. Das, A. Saha, and N. S. N. Bayev, "USenSewer: Ultrasonic Sensor and GSM-Arduino based Automated Sewerage Management," International Conference on Current Trends in Computer, Electrical, Electronics and Communication (CTCEEC 2017), 8-9 September 2017, Mysore, India.

[17] P. Dhananchezhiyan, S. S. Hiremath, M. Singaperumal, and R. Ramakrishnan, "Design and Development of a Reconfigurable Type Autonomous Sewage Cleaning Mobile Manipulator," Procedia Engineering, vol. 64, pp. 1464–1473, 2013

[18] J. Nienhuis, C. de Haan, J. Langeveld, M. Klootwijk, and F. Clemens, "Assessment of detection limits of fiber-optic distributed temperature sensing for detection of illicit connections," Water Science and Technology, vol. 67, no. 12, pp. 2712–2718, Jun. 2013

[19] M. Pereira, "Flow meters: Part 1," IEEE Instrumentation & Measurement Magazine, vol. 12, no. 1, pp. 18–26, Feb. 2009.

[20] J.-A. Maxa, M. Slim Ben Mahmoud, and N. Larrieu, "Survey on UAANET Routing Protocols and Network Security Challenges," *Ad Hoc & Sensor Wireless Networks*, vol. 37, no. 1–4, pp. 231–320, Aug. 2017.

[21] N. A. Alrajeh, S. Khan, J. Lloret, and J. Loo, "Secure Routing Protocol Using Cross-Layer Design and Energy Harvesting in Wireless Sensor Networks," *International Journal of Distributed Sensor Networks*, vol. 9, no. 1, p. 374796, Jan. 2013.

[22] S. Bhattacharjee and S. Bandyopadhyay, "An Energy Efficient-delay Aware Routing Algorithm in Multihop Wireless Sensor Networks," *Ad Hoc & Sensor Wireless Networks*, vol. 43, no. 1–2, pp. 1–32, Feb. 2019.