An Energy-Efficient LoRa IoT System for Water Monitoring: Lessons Learned and Use Cases

Seyed Alireza Rahimi Azghadi*, Kamyab Aghajamali†, Monica Wachowicz*‡, Francis Palma§, Ian Church¶, Hung Cao*

* *Analytics Everywhere Lab, Faculty of Computer Science, University of New Brunswick, Canada*

† *Offsite Construction Research Center (OCRC), University of New Brunswick, Canada*

‡ *RMIT University, Australia*

§ *SE+AI Research Lab, Faculty of Computer Science, University of New Brunswick, Canada*

¶ *Ocean Mapping Group, University of New Brunswick, Canada*

***Abstract*—With the growing impact of urbanization, industrialization, and climate change, monitoring water bodies like lakes, rivers, and reservoirs has become essential for detecting pollutants, managing resources, and preventing environmental hazards. Water monitoring is a critical process that involves continuously assessing water characteristics to ensure the safety and sustainability of water resources. We can develop detection and prediction systems by analyzing the collected data from water bodies over time. However, developing and deploying a reliable water monitoring framework must be investigated and researched. Therefore, this paper presents a water monitoring framework to address the power consumption challenge, which is one of the most important aspects of a remote monitoring system. We designed and implemented a real-time remote water monitoring framework to achieve cost and battery efficiency. We deployed our prototype in a real-world situation and discussed our challenges and limitations. Our system contains three main parts: first, an end device, which is in charge of collecting information from water and transferring the data to the land; second, a gateway placed on shore to connect the end devices to a centralized cloud server; and Finally, a cloud platform for gathering, storing and analyzing the sensor's information. Also, we provided a comprehensive study on power consumption and different end device architectures to reduce power consumption. Finally, we described different use cases of a water monitoring system using low-cost sensors.**

***Index Terms*—Marine information monitoring; Internet of Thing (IoT); Wireless sensor network (WSN); Long Range (LoRa); LoRaWAN; Smart city;**

I. Introduction

In recent decades, the rapid advancement of computer systems and the Internet of Things (IoT) has given rise to the concept of smart cities, which aim to enhance urban living quality [1]. A key feature of smart cities is their ability to monitor environmental conditions and provide data-driven insights for future decision-making. This monitoring encompasses both terrestrial and marine environments [2].

Thus, monitoring the marine environment offers valuable insights. These insights can be helpful for both governments and citizens, such as enabling the detection and prediction of environmental anomalies or informing the public about water quality for recreational activities like swimming and fishing. Consequently, developing and implementing water quality monitoring systems in urban areas can significantly benefit governmental decision-making and public welfare. However, to ensure the effectiveness of these monitoring efforts, it is essential to address several challenges and limitations. These challenges can be categorized into technological hurdles, public awareness issues, safety concerns, and resource constraints.

Several studies have been done on water monitoring, and these research has contributed to the field; however, they have several shortcomings. For instance, Hemdan et al. [3] proposed an efficient IoT-based smart water quality monitoring system. Although, the authors have done experiments in a laboratory environment without deploying the system in real-world conditions. This limitation is significant because real-world deployment is crucial to identifying challenges often overlooked in controlled environments. Additionally, the designed experiment overlooked critical aspects such as communication protocols and power consumption, which are essential for applying remote sensing technologies in water quality monitoring.

Similarly, Chen et al. [4] introduced an intelligent water monitoring system for smart cities, demonstrating cost-efficiency and real-time alerts for abnormal water quality readings. However, the study was conducted using a scaled model, which restricted exploring challenges that arise in actual environments. The use of Wi-Fi and ESP32 modules, while effective for short-distance communication, is not suitable for large-scale, real-world applications. Furthermore, Huang et al. [5] developed a practical marine wireless sensor network monitoring system utilizing LoRa and MQTT protocols, highlighting the system's long-range communication capabilities. Similarly, The study neglected to assess power consumption and battery life. Lastly, Sendra et al. [6] proposed a LoRa-based network for water quality monitoring in coastal areas, which included multiple sensors for real-time data collection. Although the study advanced the field by evaluating LoRa coverage and range, it fell short in addressing the challenges posed by environmental conditions. Similar to Huang et al. [5], the authors did not discuss the importance of energy consumption on the system's functionality over time.

The previous works on smart water monitoring systems have made significant contributions but exhibit notable shortcomings. Some studies were limited to controlled environments or scaled models, failing to capture the complexities and

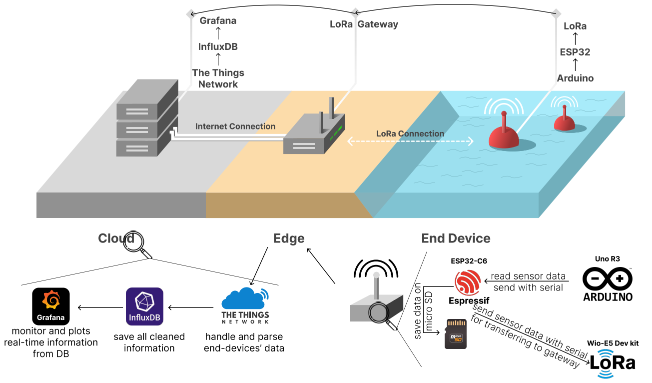


Fig. 1: Our framework architecture, illustrating different elements of the system and their connections

challenges of real-world deployment. Additionally, critical aspects like communication protocols and power management were often overlooked. These limitations highlight a gap in the existing research: the need for a comprehensive water monitoring system that not only addresses real-world environmental challenges but also optimizes communication efficiency and power consumption. In this work, we aim to fill this gap by designing and deploying a robust IoT-based water monitoring framework that is tested in real-world conditions, focusing on long-term sustainability, power efficiency, and reliable communication across various environmental settings.

Our contributions in this work are as follows:

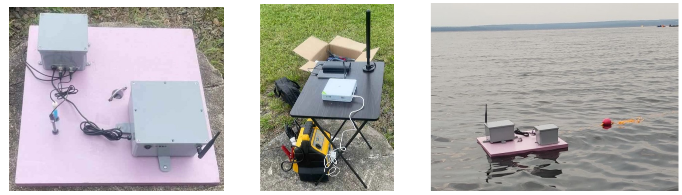
- **A Water Monitoring Framework**: We propose an IoT-based water quality monitoring framework designed to operate efficiently in diverse environmental conditions.
- **Evaluation of Real-World Deployment Challenges**: We conduct an evaluation of the challenges associated with deploying the proposed framework in real-world scenarios.
- **Power Consumption Analysis**: We perform a comprehensive analysis of the power consumption of the proposed framework, optimizing it for long-term sustainability.

In this paper, first, we discussed our designed system in Section II, explaining different parts of our system. Then, we explain two sets of experiments that we conducted in Section III. In Section IV, we discussed different scenarios in which we can employ our prototype. Finally, we summarized our work in Section V.

II. Proposed Method & Implementation

In this section, we detail the components of our framework, which was designed to be a low-cost, energy-efficient, and scalable solution for water quality monitoring. We tested our framework to prove its practicality in a real-world scenario, which is explained in Section III-2. The framework consists of several key components, as illustrated in Figure 1.

The first component was the end device, which was deployed on the water to collect data with various sensors. Figure 2a shows the end device prototype, and Figure 2c shows the end device deployment on the water. Collected



(a) End device prototype (b) Gateway (c) Deployed prototype on the water

Fig. 2: Long-Range IoT Water Monitoring Prototype and Deployment

data was transmitted via a LoRa module to a gateway located onshore. The gateway served as an intermediary, facilitating communication between the end devices and the cloud server. Figure 2b shows the LoRa gateway placed on the shore. The final component of our framework was a cloud server that executed multiple applications, enhancing the framework's scalability. In the subsequent sections, we described each component in detail.

A. End Device

To create an energy-efficient framework, we designed the end device prototype, which relied only on a small battery for extended periods as its only source of power. A key objective in our design was to ensure a long connectivity range, which led us to use LoRa technology. LoRa not only reduced power consumption but also enabled long-range communication, which made it ideal for our requirements and goals.

To further minimize power consumption, we chose the ESP32 microcontroller as the main controller for the end device. The ESP32 is known for its low energy consumption and features a deep sleep mode, which significantly reduces overall power usage. In deep sleep mode, the ESP32 powers down the main CPU and most peripherals, leaving only essential components like the Real-Time Clock (RTC) and Ultra Low Power (ULP) coprocessor active. This state allows the device to draw as little as 10 μA of current, making it ideal for battery-operated applications that require extended operational life. By using deep sleep mode, the ESP32 can wake up periodically to perform tasks such as data collection or communication and then return to sleep, thus conserving energy. This feature is particularly beneficial in IoT applications, remote sensors, and other scenarios where power efficiency is critical.

The proposed framework's end device comprises sensors and a controller responsible for transmitting water quality data to the gateway. To ensure cost-effectiveness, we selected low-cost sensors, which typically have short wires and lack attachments. These short wires restrict the placement of the end device to areas near the water surface, while the controller must be positioned in a convenient location on a floating platform (such as a boat or buoy) for deployment and development. Additionally, the end device's communication module must be elevated to transmit data to a gateway that is placed at a long distance. To address this challenge, we decoupled the

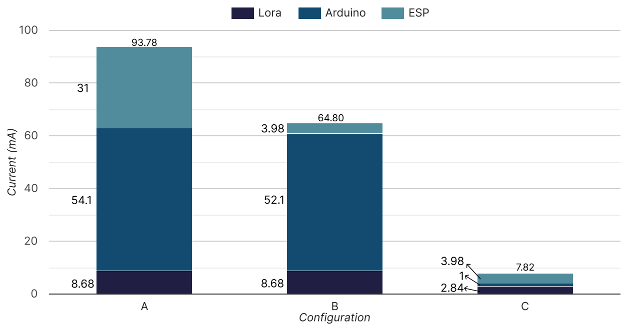


Fig. 3: Different prototype configuration analysis at 1 minute sampling rate

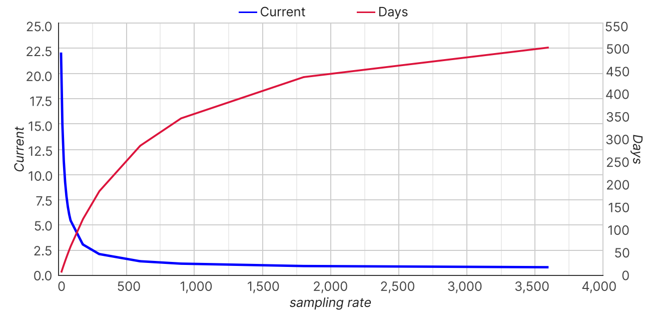


Fig. 4: End device current draw and maximum operating hours with respect to sampling rate

controller from the data acquisition system. We employed an Arduino for data acquisition, which provides ample analog and digital input channels for seamless integration of additional sensors. By placing the Arduino and sensors in a box near the water surface and positioning the transmitter module at an elevated location, we achieved greater flexibility in deploying the end device and enhanced adaptability to various low-cost sensors available on the market.

B. Gateway

We installed a LoRa gateway near the water bodies to establish communication between the end devices and the cloud server. Each LoRa gateway can support up to a thousand connections from various end devices and can provide a line-of-sight range of up to 10 kilometres when equipped with an appropriate antenna. The gateway was connected to the internet, enabling it to route the received data packets from each end device to our cloud server. We used SenseCAP M2 Multi-Platform LoRaWAN Indoor Gateway. We used its web interface to program the gateway to forward the received packets to our cloud server. The mentioned LoRa gateway is cheap and suitable for our experiment. However, it is unsuitable for long-term deployment as it does not have IP67 standards to resist rain and harsh weather.

C. Cloud Server

We used The Things Network (TTN) as the primary tool for managing the large volumes of data collected from multiple gateways and sensors. TTN was responsible for authorizing gateways and end devices, preventing malicious users from injecting false data into our system. Each end device and the gateway was first registered in our system, and after registration, acknowledgments were periodically sent to the server, which showed their connectivity. Whenever TTN receives new data from an authorized device, TTN parses the compressed data received from the gateway into a human-readable format. This data is then forwarded to an InfluxDB instance running on the server. InfluxDB is an SQL database optimized for handling substantial amounts of time-series data and offers various extensions for retrieving data into other applications. To provide real-time monitoring and insights into the stored data, we deployed an instance of Grafana, a robust monitoring tool. Grafana offers a range of features that allow users to create customized dashboards and set alerts based on specific conditions. All communication between these components occurs within a secure network inside a Docker environment on the cloud server, enhancing the overall security of the framework.

III. Experiments

1) Power Consumption Analysis: We implemented four versions to evaluate power consumption and calculate how long our prototype can operate with a standard 10,000 mAh battery. Initially, we used a Raspberry Pi 4B as the main controller because it is more convenient for developing code and attaching peripherals. However, the average current draw of the controller limited the prototype to under 10 hours of operation. The Raspberry Pi draws approximately 0.9 A, which is excessive for an IoT project. In our second attempt, we replaced the Raspberry Pi with an ESP32C6.

Besides the Raspberry Pi configuration, we illustrated the current draw of different modules for the other three configurations in Figure 3. In Configuration A, an ESP32C6 is used as the controller, resulting in a current draw of 93.078 mA. Configuration B employs the deep-sleep mode in both the ESP32 and Arduino modules, reducing the current draw to 64.80 mA, representing a 43% improvement in power efficiency compared to Configuration A. Configuration C utilizes the ESP32's deep-sleep mode and turns off the Arduino and LoRa modules when they are inactive, further reducing the current consumption to 7.82 mA. This setup demonstrates a 728% improvement in battery life compared to Configuration B. We measured the current draw of the different configurations with a 1-minute sampling rate.

To calculate the operation hours of the prototype based on the measured current draw, we used Equation 1. Each battery has a capacity stated on it. Also, batteries produce different voltages, but our system works with 5 V. So, in Equation 1, we converted the capacity based on the battery's output voltage and then calculated total operation works with the average current draw of our circuit.

$$\text{Time}(h) = \frac{\text{Battery Capacity}(Ah) \times \text{Battery Voltage}(V)}{\text{Circuit Current}(A) \times \text{Circuit Voltage}(V)} \tag{1}$$

Finally, when we found our best configuration, we calculated different operation hours regarding the sampling rate. Because we turned off two modules in our prototype, we conserved more energy when not sampling the water with our sensors. So, we calculated the maximum sampling rate that allows our prototype to work for about one year with a 5Ah battery. We illustrated our finding in Figure 4. In Figure 4, the blue line shows the circuit current draw regarding the sampling rate, and the red line shows the maximum operation hours with a specific sampling rate with a 4500mA (11.1V) battery. If we sample the water body every 15 minutes, our system is expected to work for 354 days.

2) Lessons Learned: This section explained some challenges we faced during our experiment in Grand Lake, New Brunswick, Canada. We took our prototype to the lake to test the whole system's performance for a short period of time. We collected data for several hours and observed the prototype during the sampling.

One of the primary challenges in implementing a water monitoring system on a buoy is ensuring that the platform remains securely anchored in place, as wind and tidal forces can easily cause it to drift to unintended locations. Developing a solution that keeps the system fixed in a designated area of the lake is essential; also, adding a GPS module to the prototype can contribute to the quality of collected data if the end device's precise location is important. Another significant concern is the potential for interference from passing watercraft, such as jet skis and boats. During our experiment in Grand Lake, shown in Figure 2c, multiple curious jet skies were lingering around the installed buoy, posing a potential risk of damaging the system. Curious individuals may tamper with the system, disrupting its performance by pulling sensors out of the water or manipulating wired connections. Additionally, designing a reliable floating structure is crucial, as the platform must remain buoyant on the water. This also involves constructing a waterproof container to protect the system's sensitive components, presenting further technical challenges.

IV. Use Cases

In this paper, we proposed a general water-monitoring framework considering different goals to achieve. In the following section, we provided examples of use cases in our framework, which require a water monitoring system that can remotely collect sensor information from long distances and operate for a long time.

A. Flood Monitoring

Our prototype can be installed upstream in rivers to monitor water levels, providing early warnings to decision-makers before a flood occurs. Offering real-time data can help mitigate the impact of floods, significantly reduce potential damage to agriculture and safeguard lives in areas prone to flooding [7].

B. Green Blue Algae

Another potential application of our prototype is the detection of blue-green algae, also known as cyanobacteria, a harmful bacterium to both humans and animals. According to Rocher et al. [8], cyanobacteria can be detected using low-cost turbidity sensors. These sensors can be integrated into our prototype, enabling the collection and monitoring of data to identify the presence of blue-green algae effectively. This functionality could enhance environmental monitoring efforts by providing a cost-effective solution for early detection of harmful algal blooms.

C. Water Pollution Management

Another use case for this tool is in water pollution management. Since sewage and other contaminants can enter open rivers, integrating sensors that detect heavy metals and monitor pH levels into the system can help identify pollution. This allows for timely contamination detection, enabling decision-makers to take action and remove pollutants from the water, ensuring safer and cleaner waterways.

V. Conclusion and Future Work

We designed and implemented a remote water monitoring framework aiming to build and test a prototype capable of collecting data from long distances and operating for an extended time. Also, We tested our prototype in Grand Lake, NB, Canada, and discussed real-world deployment challenges, which are not obvious at first sight. Besides, we provided a comprehensive analysis of power consumption. We tested several architectures to achieve a prototype that can operate for about a year with a high sampling rate. Finally, we discussed some possible applications of our system. In the future, we want to analyze the system from different aspects. First, we will evaluate the system's long-term performance. The second aspect of our framework that needs to be tested is scalability. Finally, we plan to gather datasets from various locations over a sufficient period of time to capture multiple marine cycles.

Acknowledgment

This work was supported by Ocean Frontier Institute Seed Fund and New Brunswick Innovation Foundation (NBIF EP2024-003).

References

[1] Zhihan Lv et al. Infrastructure monitoring and operation for smart cities based on iot system. *IEEE Transactions on Industrial Informatics*, 16:1957–1962, 2020.

[2] Murat Bakirci. Smart city air quality management through leveraging drones for precision monitoring. *Sustainable Cities and Society*, 106:105390, 2024.

[3] Ezz El-Din Hemdan et al. An efficient IoT based smart water quality monitoring system. *Multimedia Tools and Applications*, 82(19):28827–28851, 2023-08-01.

[4] Shih-Lun Chen et al. An intelligent water monitoring IoT system for ecological environment and smart cities. *Sensors*, 23(20):8540, 2023-01. Number: 20 Publisher: Multidisciplinary Digital Publishing Institute.

[5] Ao Huang et al. A practical marine wireless sensor network monitoring system based on LoRa and MQTT. In *2019 IEEE 2nd International Conference on Electronics Technology (ICET)*, pages 330–334, 2019-05.

[6] Sandra Sendra et al. LoRa-based network for water quality monitoring in coastal areas. *Mobile Networks and Applications*, 28(1):65–81, 2023.

[7] M. Cho et al. Water level prediction model applying a long short-term memory (lstm)–gated recurrent unit (gru) method for flood prediction. *Water*, 2022.

[8] Javier Rocher et al. Development of a low-cost optical sensor to detect eutrophication in irrigation reservoirs. *Sensors (Basel, Switzerland)*, 21, 2021.