An Energy Efficient LoRa-based Multi-Sensor IoT Network for Smart Sensor Agriculture System

Shivakant Mishra
Department of Computer Science
University of Colorado Boulder
CO, USA
mishras@colorado.edu

Sanjeet Nayak
Department of CSE
IIITDM Kancheepuram
Chennai, India
sanjeetn@iiitdm.ac.in

Ramnarayan Yadav
Department of Electrical & Computer Science Engineering
IITRAM
Ahmedabad, India
ramnarayanan@iitram.ac.in

2023 IEEE Topical Conference on Wireless Sensors and Sensor Networks (WiSNeT) | 978-1-6654-9321-5/23/$31.00 ©2023 IEEE | DOI: 10.1109/WISNET56959.2023.10046242

***Abstract*— With the advancement and usage of Internet of Things (IoT), smart agriculture is evolving rapidly. Smart solutions based on IoT can not only help the farmers in maximizing the profits but also help them in reducing the manual supervision of agriculture land. In a smart agriculture system, inexpensive, resource-constrained sensors are installed near the crops as well as at some strategic locations in an agriculture field to collect relevant crop and environment data in real time. This data is then used for both critical, latency-sensitive decision making as well as long-term planning. The key challenges in building a smart agriculture system include high communication latency and bandwidth consumption incurred with computing on the cloud, frequent Internet disconnections in rural areas, and a need for keeping the overall cost low for the end users (farmers). In this paper, we discuss the design and implementation of our on-going, LoRa-based three-tier smart agriculture system comprised of (i) Sensing layer, (ii) Fog layer, and (iii) Cloud layer. In particular, we focus on the how the low-power, low-bandwidth and long-range features of LoRa are utilized to transform traditional agriculture land of rural areas in India into smart agriculture system. We present the performance of our current prototype and compare with the existing, state-of-the-arts framework for smart agriculture system in terms of cost, latency and distance.**

Keywords*— smart agriculture, edge and fog computing, LoRa, low-cost sensors, performance

I. Introduction

Smart agriculture is one of the most promising areas where IoT-enabled technologies have the potential to substantially improve the quality and quantity of the crops and reduce the associated operational cost. At present, cloud-based infrastructures are being utilized to support various smart agriculture applications and data processing. Data from smart sensors in the agricultural field is transmitted to the cloud over the Internet, and then stored and processed in the cloud for decision making. While cloud-based infrastructures certainly offer enormous processing power and storage capacity, there are two key limitations that need to be addressed when used in the context of smart agriculture [1]: (i) Sensor data transmitted over the Internet requires continuous Internet connectivity, consumes high bandwidth and incurs delays. (ii) Since IoT devices must transmit large volumes of data to the cloud for storage and processing, the energy of battery-powered IoT devices is quickly be drained. These limitations make cloud-based infrastructure ill-suited for smart agriculture. To address these limitations, we propose a LoRa-enabled, fog-based smart agriculture infrastructure that will reduce the quantity of data transferred to the cloud and enable latency-sensitive services delivered just in time.

This paper provides a high-level overview of our project whose goal is to develop an end-to-end, LoRa-enabled, fog-based infrastructure for smart agriculture in India and USA. We focus on the how the low-power, low-bandwidth and long-range features of LoRa are utilized to transform traditional agriculture lands in rural areas into smart agriculture system. We describe the design and a preliminary implementation of a microservice-based edge server that allows to provide important, latency-sensitive services to the farmers and enable operation in a disconnected Internet environment. We present the performance of our current prototype and compare with the existing, state-of-the-arts framework for smart agriculture system in terms of cost, latency and distance.

II. System Architecture

The architecture of the proposed microservice-based fog enabled infrastructure for smart agriculture is shown in the Figure 1. It consists of the following layers: sensing layer, fog computing layer, and cloud computing layer, which are linked by cross-layer upstream and downstream communication for data and control information flows.

The sensing layer is comprised of the sensors and actuators deployed across the agricultural field to periodically sense the physical parameters of interest such as air temperature, air humidity, solar radiation, soil temperature and moisture at various depths, wind speed and rainfall. In a typical scenario, agriculture fields tend to be at remote locations with low-bandwidth and poor Internet connectivity. Furthermore, farmers need to remotely monitor large agriculture fields spanning distances of 5 km to 10 km, or even more. To address these challenges, we have adopted a LoRa and LoRaWAN enabled communication system due to their support for low power, wide area networking designed to wirelessly connect limited energy operated IoT devices to an edge server at a distance of 1-2 km.

Authorized licensed use limited to: UNIVERSIDADE FEDERAL DO PIAUI. Downloaded on July 30,2023 at 13:14:42 UTC from IEEE Xplore. Restrictions apply.

The fog computing layer is comprised of one or more servers, e.g. a Dell PowerEdge or Dell Precision server. This layer provides an administrative control of the entire IoT infrastructure of the agricultural field. This layer addresses the limitations of intermittent Internet connectivity, high latency and high network bandwidth consumption of cloud-based infrastructure. The fog nodes collect data from the sensors deployed in the field, perform key data cleaning, filtering, aggregation and fusion tasks, and execute latency sensitive services, such as fire management and animal intrusion detection. To facilitate a flexible architecture that may utilize existing container-based support for various ML/AI services, we have structured the fog layer as a microservice architecture [2]. In this architecture, an application is composed from a collection of loosely-coupled microservices, where each microservice is fine-grained and the associated protocols are lightweight. Benefits of this architecture include modularity, scalability, software reuse, and integration. In our prototype, we are employing some of the latest state-of-the-art methodologies for container instantiation and intercontainer communication at the layer for high efficiency [3].

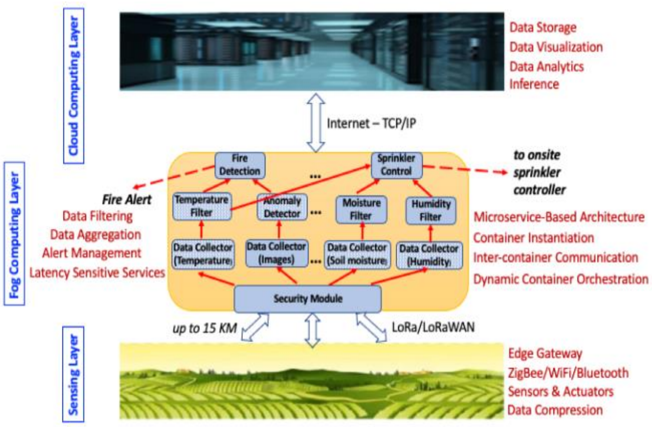

Figure 1: End-to-end Architecture of a LoRa-Enabled, Fog-Based Infrastructure for Smart Agriculture Smart Agriculture

Finally, the cloud computing layer is used for performing complex, computational intensive decision making as well as for long-term data storage and visualization.

III. Sensing Layer

Sensing layer consists of several End nodes whose task is to sense the agriculture field parameters and communicate them to Fog layer using LoRa communication protocol. The end node is responsible for sending the sensors' data to fog node or receiving the actions/commands for actuation from fog nodes.

First, we build an End node (shown in Figure 2) which consists of various sensors (required for smart agriculture), one microcontroller (e.g., ESP32, Arduino Mega, etc.), LoRa Hat with Antenna. LoRa hat is build using LoRa SX1276 IC.

Various sensors like PIR, IR, Capacitive soil moisture, Humidity and Temperature are connected to Arduino microcontroller that monitors motion, moisture content in the soil, humidity and temperature of the environment respectively. Table I shows sensor's accuracy used in our implementation.

Figure 2: LoRa End node

Table I: Different types of sensors

Sensors	Accuracy and Range
Humidity and Temperature (DHT11)	Humidity range: 20%~90% Temp. range: 0~+50°C
IR Sensor (LM393)	distance: 2~30cm Detection angle: 35°
PIR (ZRD09)	distance: 4~8m Detection angle: 100°
Capacitive Soil Moisture	Operating voltage:3.3 ~ 5.5V

As a part of our first experiment, we demonstrated how a fog node can connect and/or communicate multiple end nodes. We used single channel (it can communicate with one end node at a time) LoRa antenna at fog node. To experiment the scheduling capability of fog node, we connected three end nodes with one LoRa enabled gateway (Raspberry Pi with LoRa hat) as shown in Figure 3. The end nodes are scheduled in a round robin fashion by fog node to avoid interference of data during simultaneous communication by the three end nodes.

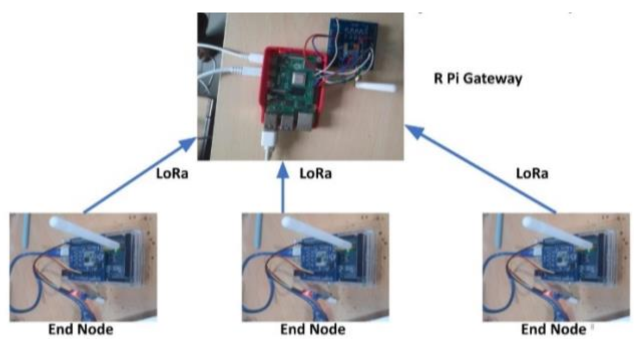

Figure 3: LoRa communication using multiple End nodes

Table II: LoRa experiment result

Sender / Receiver Node (Height, Distance)	Latency
Sender: 4ft, Receiver: 4ft, Sender ~ Receiver: 500m	4.0s
Sender: 4ft, Receiver: 35ft, Sender ~ Receiver: 350m	0.7s
Sender: 35ft, Receiver: 200ft, Sender ~ Receiver: 500m	1.1s
Sender: 4ft, Receiver: 200ft, Sender ~ Receiver: 2000m	0.9s

Table II shows the experimental results obtained while communicating data from end node to the fog gateway. We

Authorized licensed use limited to: UNIVERSIDADE FEDERAL DO PIAUI. Downloaded on July 30,2023 at 13:14:42 UTC from IEEE Xplore. Restrictions apply.

varied the height of sender and receiver and also the distance between them and checked the delay in LoRa communication. We performed these experiments in environment where are objects and building were present (these obstacles cause channel attenuation thus affect the received signal strength). Data that is communicated between sender and receiver is 16 bytes. It can be noticed from the Table II that communication latency depends distance and height of communicating nodes. It can be observed that as we increase distance between sender and receiver, latency increases. Latency can be reduced further by placing receiver (fog node) at proper height for real time application such as Fire detected or animal intrusion detection.

The LoRa communication technology (used in this work) is compared with existing other technologies like Zigbee, WiFi, Bluetooth low energy (BLE) is provided in Table III. LoRa communication has high range (upto 15 KM in rural areas) and it consumes very less power, which makes it suitable for application areas where Internet connection is intermittent.

Table III: Comparison of different communication technology

Technology	**LoRa**	**Zigbee**	**WiFi**	**BLE**
Standard	LoRaWAN	IEEE 802.15.4	IEEE 802.11 b/g/n	IEEE 802.15.4
Frequency	868 MHz 915 MHz	2.4 GHz	2.4 GHz 5 GHz	2.45 GHz
Topology	Star	Mesh	Star	Mesh
Range	2 - 5 KM (Urban) 10 -15 KM (Rural)	10-100 M	5-10 M	5-10 M
Data Rate	3 – 50 Kbps	250 Kbps	150 Mbps	2 Mbps
Power Consumption	Very Low	Very Low	Low	Low

IV. FOG LAYER

The goal of fog layer is to provide support for latency-sensitive services as well as operation in a disconnected environment. In order to gain a deeper understanding and better motivate the design of our proposed smart agriculture application, we first need to learn the ground reality of the key challenges faced by farmers in agriculture. To this end, we conducted a survey with five farmers in India to understand (1) what are the key issues they face that could be addressed by smart agriculture? (2) what tasks could be simplified by smart agriculture? and (3) what new and useful features could be provided by smart agriculture that are infeasible at present? Top three challenges identified by the farmers are (1) animal intrusion in the field, particularly in the night; (2) rising manual labor cost; and (3) fire hazards and efficient irrigation.

Our goal is to address each of these challenges. At present, we are addressing the problem of animal intrusion. In particular, our goal is to automatically detect animal intrusions, develop an actuation method to drive away the animals, and inform the farmer(s) in a timely manner about the intrusion. Figure 4 shows an architecture diagram. Since this is a latency sensitive service, it is completely implemented in the Fog computing layer. We are experimenting with three types of sensors, a PIR motion sensor that detects if there is any movement in its field of view, an ultrasonic sensor that detects distance of the object, and an infrared camera sensor that creates an an image using infrared radiation. Details of what actuation we can provide in case an animal intrusion is detected is currently being investigated. The current actuation mechanism is switching on an LED light.

The Security Module passes the sensor data it receives from authenicated sensors to the appropriate Data Collector & Filter container (Distance, IR images or Motion). Data Collection & Filter containers perform two tasks. First, they drop any spurious data, e.g. a distance value larger than 15 meters, or an anomalous IR image. Second, they store copies of the data received in the last ten minutes or so (time parameter is configurable). Finally, the Animal Detection container detects if there is an animal intrusion, including perhaps identifying the type of animal using machine learning models.

The Animal Detection container is being designed to operate for any or all of the three sensors. For example, based on the cost-benefit tradeoff, a farmer may choose a low-cost solution and install only PIR motion sensors, in which case the accuracy of animal intrusion would be lower, while another farmer may choose to install all three types of sensors, in which case the accuracy of animal intrusion would be much higher.

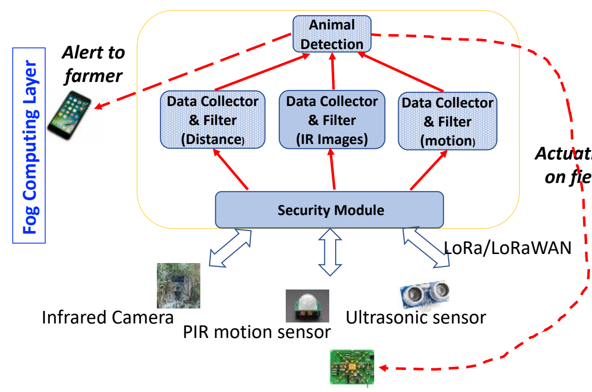

Figure 4: End-to-end Architecture of a LoRa-Enabled, Fog-Based Infrastructure for Smart Agriculture Smart Agriculture

All intercontainer communication at this layer, e.g. from security module to Data Collector & Filter containers or fromm Data Collector & Filter containers to Animal Detection container is implemented using our shared memory based inter-container tool that is compatible with Kubernetes [3, 4].

V. FUTURE WORK

This project is currently on-going with a goal to develop an end-to-end prototype of a smart agriculture system from low-cost sensors on one end to the cloud on the other end with edge server in the middle.

ACKNOWLEDGMENT

This research is supported in part by grants from TIH-IoT (TIH-IoT/2022-07/ IC /NSF/SL/ NIUC-2022-04/005) and NSF (13012641).

Authorized licensed use limited to: UNIVERSIDADE FEDERAL DO PIAUI. Downloaded on July 30,2023 at 13:14:42 UTC from IEEE Xplore. Restrictions apply.

References

[1] W. Shi, J. Cao, Q. Zhang, Y. Li and L. Xu. “Edge computing: Vision and challenges”. IEEE Internet Things”, J. 2016;3:637–646. doi: 10.1109/JIOT.2016.2579198.

[2] R. Chen, S. Li and Z. Li. “From Monolith to Microservices: A Dataflow-Driven Approach,”, 24th Asia-Pacific Software Engineering Conference (APSEC), 2017.

[3] M. Abranches, S. Goodarzy, M. Nazari, S. Mishra and E. Keller. “Shimmy: Shared Memory Channels for High Performance Inter-Container Communication”. In The 2nd USENIX Workshop on Hot Topics in Edge Computing (HotEdge 2019), Renton, WA (July 2019).

[4] M. Khasgiwale, V. Sharma, B. Thadachi, S. Mishra, J. John and R. Khanna. “Shimmy: Accelerating Inter-Container Communication Through Shared Memory for the IoT Edge”. Submitted for publication.

Authorized licensed use limited to: UNIVERSIDADE FEDERAL DO PIAUI. Downloaded on July 30,2023 at 13:14:42 UTC from IEEE Xplore. Restrictions apply.